

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

اللَّهُ نُورُ السَّمَوَاتِ وَالْأَرْضِ

خدا یا چنان کن سرانجام کار
تو خوشنود باشی و ما رستگار



دانشکده علوم ریاضی

پایان نامه کارشناسی ارشد ریاضی کاربردی

کدهای پوششی و کران‌های بالا روی ابعاد بازی‌های ساده

نگارنده: راضیه مولایی

استاد راهنما

دکتر میثم علیشاهی

استاد مشاور

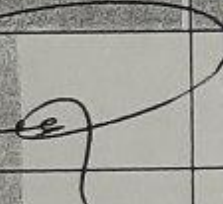
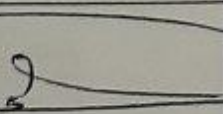
دکتر کاظم خشیارمنش

دی ۱۳۹۸

فرم شماره (۳) صورتجلسه نهایی دفاع از پایان نامه دوره کارشناسی ارشد

با نام و یاد خداوند متعال، ارزیابی جلسه دفاع از پایان نامه کارشناسی ارشد خانم راضیه مولایی با شماره دانشجویی ۹۶۱۴۴۷۴ رشته ریاضی کاربردی گرایش تحت عنوان کدهای پوششی و کران های بالا روی ابعاد بازی های ساده که در تاریخ ۱۳۹۸/۱۰/۰۲ با حضور هیأت محترم داوران در دانشگاه صنعتی شاهرود برگزار گردید به شرح ذیل اعلام می گردد:

- الف) درجه عالی: نمره ۱۹-۲۰ ☐ ب) درجه خیلی خوب: نمره ۱۸-۱۸/۹۹ ☒
ج) درجه خوب: نمره ۱۶-۱۷/۹۹ ☐ د) درجه متوسط: نمره ۱۴-۱۵/۹۹ ☐
ه) کمتر از ۱۴ غیر قابل قبول و نیاز به دفاع مجدد دارد ☐
نوع تحقیق: نظری ☒ عملی ☐

عضو هیأت داوران	نام و نام خانوادگی	مرتبه علمی	امضاء
۱- استاد راهنمای اول	دکتر میثم علیشاهی	دانشیار	
۲- استاد راهنمای دوم	-----	-----	-----
۳- استاد مشاور	دکتر کاظم خشیارمنش	استاد	عائیه
۴- نماینده تحصیلات تکمیلی	دکتر سمیه مغاری	استادیار	
۵- استاد ممتحن اول	دکتر عبدالله آل هوز	دانشیار	A.A
۶- استاد ممتحن دوم	دکتر ابوالفضل پورعیدی	استادیار	

نام و نام خانوادگی رئیس دانشکده: دکتر مهرداد قهرمانمند

تاریخ و امضاء و مهر دانشکده:



تبصره: در صورتی که کسی مردود شود حداکثر یکبار دیگر (در مدت مجاز تحصیل) می تواند از پایان نامه خود دفاع نماید (دفاع مجدد)

نقدیم به مقدس‌ترین واژه‌ها در لغت‌نامهٔ دلم،
پدر بزرگوارم و مادر مهربانم...

امروز
هستی‌ام
به
امید
شماست
و
فردا
کلید
باغ
بهشتم
رضای
شما.



هو العليم،

سپاس خداوندگار حکیم را که با لطف بی کران خود، آدمی را زیور عقل آراست.

زیباترین نام را بر زبان جاری می کنم...

که هر کس زبان به حمد تو گشود بی تردید نگاه تو بر او افتاده. پس بر قلمم آن جاری کن که خود می پندی در ثنایت لب کشایم.
 در وادی معرفت گنج، سرچشمه هدایت نبو شد، سرب قامت بندی فرو نیاید، گر گنجینه ای را که مقدسش خواندی و به آن قسم یاد
 کردی اگوچک شمرده شود و تنها خاطره جوهر خشک شده ای از آن بر برگ برگ صفحات زندگی باقی ماند.
 تو علم را روشنی قرار دادی و فانوسی در پیغوله راه که مسیر را، راه نماید و تزییه را مقدم بر آن دانستی تا نگهبانش باشد که تزییه و تعلیم در
 معیت هم، کو هر وجودی انسان را به نور تو منور کند، پرده از واقعیات کنار زند. آن جاست که حقیقت رخ نمایاند، نظر فراتر افتد،
 خوان گنجینه های دانش رنگین شود...

آری آن جاست که آدمی معنا یابد.

من اگر وعده یام با تو زیر خروار مائل فراموشی و غفلت مدفون گردیده، اگر زشتی ظفیان در نظرم زیبا جلوه گری می کند و چشمانم
 خشک تر از آن است که در مقام توبه اسکی بر آن جاری شود، بدان از سر جهل است و نسیان...
 آبا بار الهام چشم طمع بر رحمت دوخته ام و در تمنای ربانی از ظلمت ضلالت، ترنم باران معرفت را می طلبم، امید آن که جوانه های
 حقیقت را در وجودم برویاند و انعکاس آن چشمانم را روشن کند.

زبان قاصراست و مجال کوتاه...

تو خود قصیده ی مهر را از لوح نانوشتی قلمم بخوان...



سپاسگزاری

«مَنْ لَمْ يَشْكُرِ الْمَخْلُوقَ لَمْ يَشْكُرِ الْخَالِقَ»

از آنجایی که تجلیل از معلم، سپاس از انسانی است که هدف و غایت آفرینش را تأمین می‌کند؛ بر حسب وظیفه و از باب «مَنْ لَمْ يَشْكُرِ الْمَنْعَمَ مِنَ الْمَخْلُوقِينَ لَمْ يَشْكُرِ اللَّهَ عَزَّوَجَلَّ»:

در آغاز از زحمات بی دریغ استاد همیشه راهنمایم؛ جناب آقای دکتر میثم علیشاهی صمیمانه تشکر و قدردانی می‌کنم که در کمال سعه صدر، با حسن خلق و فروتنی، از هیچ کمکی در این عرصه بر من دریغ ننمودند، که به‌راستی اسم راهنما برازنده ایشان است و همواره شاگرد مکتب علم و انسانیت و منش والای ایشان هستم. از خداوند بهترین‌ها را برایشان آرزومندم.

از جناب آقای دکتر کاظم خشیارمنش که زحمت مطالعه و مشاوره این پایان‌نامه را تقبل فرمودند کمال امتنان را دارم.

همچنین لازم می‌دانم از اساتید فرهیخته و دلسوز؛ جناب آقای دکتر عبدالله آل‌هوز و دکتر ابوالفضل پورعیدی که زحمت داوری این پایان‌نامه را متقبل شدند با تمام وجود تشکر و قدردانی نمایم.

در پایان، بوسه می‌زنم بر دستان خداوندگاران مهر و مهربانی، دو معلم بزرگواری، پدر و مادر عزیزم؛ و بعد از خدا، ستایش می‌کنم وجود مقدس‌شان را که همواره در تمام عرصه‌های زندگی یار و یآوری بی چشم‌داشت برای من بوده‌اند.

راضیه مولایی

دی ۱۳۹۸

تعهد نامه

اینجانب راضیه مولایی دانشجوی کارشناسی ارشد رشته ریاضی کاربردی دانشکده علوم ریاضی دانشگاه شاهرود، نویسنده پایان نامه با عنوان **کدهای پوششی و کران های بالا روی ابعاد بازی های ساده**، تحت راهنمایی **دکتر میثم علیشاهی** متعهد می شوم:

- تحقیقات در این پایان نامه توسط اینجانب انجام شده است و از صحت و اصالت برخوردار است.
- در استفاده از نتایج پژوهش های دیگر پژوهش گران، به مرجع مورد استفاده استناد شده است.
- مطالب این پایان نامه، تا کنون توسط خود، یا فرد دیگری برای دریافت هیچ نوع مدرک یا امتیازی در هیچ جا ارایه نشده است.
- حقوق معنوی این اثر، به دانشگاه صنعتی شاهرود تعلق دارد، و مقالات مستخرج با نام «دانشگاه صنعتی شاهرود» یا “Shahrood University of Technology” به چاپ خواهد رسید.
- حقوق معنوی تمام افرادی که در به دست آوردن نتایج اصلی پایان نامه تاثیرگذار بوده اند، در مقالات مستخرج از پایان نامه رعایت می گردد.
- در تمام مراحل انجام این پایان نامه، در مواردی که از موجود زنده (یا بافت های آنها) استفاده شده است، ضوابط و اصول اخلاقی رعایت شده است.
- در تمام مراحل انجام این پایان نامه، در مواردی که به حوزه اطلاعات شخصی افراد دسترسی یافته (یا استفاده شده است)، اصل رازداری و اصول اخلاق انسانی رعایت شده است.

راضیه مولایی

دی ۱۳۹۸

مالکیت نتایج و حق نشر

- تمام حقوق معنوی این اثر و محصولات آن (مقالات مستخرج، کتاب، برنامه های رایانه ای، نرم افزارها و تجهیزات ساخته شده) متعلق به دانشگاه صنعتی شاهرود می باشد. این مطلب باید به نحو مقتضی، در تولیدات علمی مربوطه ذکر شود.
- استفاده از اطلاعات و نتایج موجود در این پایان نامه بدون ذکر منبع مجاز نمی باشد.

چکیده

یک شرایط با N بازیکن در نظر بگیرید که در آن بعضی از بازیکن‌ها از یک ائتلاف با یک هدف جمعی خاص می‌آیند. بازی‌های ساده برای مدل‌سازی سیستم‌ها استفاده می‌شوند و تصمیم می‌گیرند که آیا ائتلاف موفق بوده (برد) یا خیر (باخت). یک بازی ساده را می‌توان به عنوان یک تابع بولی یکنوا نگاه کرد. ابعاد یک بازی ساده، کوچک‌ترین عدد صحیح مثبت d است به طوری که بازی ساده می‌تواند به عنوان تقاطع d تابع آستانه‌ای تعریف شود؛ جایی که هر تابع آستانه‌ای از یک مقدار آستانه و وزن‌های n استفاده می‌کند. در این پایان‌نامه، کران بالا و پایین برای بُعد بازی‌های ساده مورد مطالعه قرار می‌گیرد. در این رابطه کران بالای ارائه شده توسط تیلور و زویکر ارائه خواهد شد، این کران بالا بیان می‌کند که بُعد یک بازی ساده حداکثر تعداد اعضای خانواده ائتلاف‌های ماکسیمال باخت است، سپس این کران به طور چشمگیری به تعداد اعضای یک کد پوششی دودویی با اندازه n و شعاع ۱ بهبود پیدا می‌کند. موضوع دیگری که در این پایان‌نامه مورد بررسی قرار می‌گیرد، مسئله بازی کلاه است. به کمک کدهای پوششی تعدادی از استراتژی‌های بهینه برای بازی کلاه در هر دو حالت متقارن و نامتقارن ارائه خواهد شد.

کلمات کلیدی: بُعد، ائتلاف ماکسیمال باخت، کد پوششی دودویی، حالت متقارن، حالت نامتقارن.

فهرست مطالب

ف	فهرست تصاویر
ق	فهرست جداول
ش	فهرست اختصارات
۱	۱ مفاهیم و تعاریف مقدماتی
۱	۱.۱ مفاهیم مقدماتی در نظریه کدگذاری
۷	۲.۱ قاعده‌های کدگذاری
۷	۱.۲.۱ قاعده کدگذاری ماکزیمم احتمال (MLD)
۷	۲.۲.۱ قاعده کدگذاری مینیمم فاصله (MDD)
۸	۳.۱ کران‌هایی در نظریه کدگذاری
۹	۱.۳.۱ کران‌های پایین برای کدها
۱۰	۲.۳.۱ کران‌های بالا برای کدها
۱۱	۴.۱ مفاهیم مقدماتی در نظریه رمزنگاری
۱۵	۲ کدهای پوششی و کران‌های بالا روی ابعاد بازی‌های ساده
۱۵	۱.۲ معرفی
۱۶	۲.۲ فعالیت‌های مرتبط
۱۷	۳.۲ مقدمات
۱۷	۱.۳.۲ بازی‌های ساده
۲۰	۲.۳.۲ کدهای پوششی
۲۱	۴.۲ اولین کران بالا
۲۲	۱.۴.۲ ورودی
۲۳	۲.۴.۲ گام ۱
۲۳	۳.۴.۲ گام ۲
۲۴	۴.۴.۲ گام ۳

۲۴	 خروجی	۵.۴.۲
۲۵	 اثبات اولین کران بالا	۵.۲
۲۹	 کران بالای دوم	۶.۲
۳۳	بازی کلاه	۳
۳۴	 معرفی	۱.۳
۳۶	 مجموعه‌های مناسب	۲.۳
۳۶	 معرفی مجموعه‌های مناسب	۱.۲.۳
۴۱	 مجموعه‌های مناسب در حالت کلی ($N > 1$)	۲.۲.۳
۴۵	 بازی کلاه دونفره	۳.۳
۴۶	 بازی کلاه دونفره متقارن	۱.۳.۳
۴۸	 بازی کلاه دونفره نامتقارن	۲.۳.۳
۵۰	 بازی کلاه سه‌نفره	۴.۳
۵۰	 بازی کلاه سه‌نفره متقارن	۱.۴.۳
۵۱	 بازی کلاه سه‌نفره نامتقارن	۲.۴.۳
۵۲	 بازی کلاه چهارنفره	۵.۳
۵۲	 بازی کلاه چهارنفره متقارن	۱.۵.۳
۵۳	 بازی کلاه چهارنفره نامتقارن	۲.۵.۳
۵۵	 بازی کلاه پنج‌نفره	۶.۳
۵۵	 بازی کلاه پنج‌نفره متقارن	۱.۶.۳
۵۶	 بازی کلاه پنج‌نفره نامتقارن	۲.۶.۳
۶۱	 مجموعه‌های پوششی	۷.۳
۶۲	 پیچیدگی محاسباتی	۸.۳
۶۳	 بررسی اجمالی	۹.۳
۶۵	نتیجه‌گیری و پیشنهاد	۴
۶۵	 نتیجه‌گیری	۱.۴
۶۶	 پیشنهاد	۲.۴
۶۷	آ مولد AS	
۷۱	ب ماتریس تصمیم مولد	
۷۹	پ آزمون سعی و خطا ($N = 2, p = 0.5$)	
۸۳	ت آزمون سعی و خطا ($N = 3; p = 0.5$)	

ث $N = 4$ ، لیست مرتب‌شده، $p = 9/0$ ، $nas = 40$ ، $nasopt = 24$ ۸۹

ج $N = 5$ ، $p = 9/0$ ، لیست مرتب‌شده (انتخاب‌شده). ۹۱

مراجع ۹۳

واژه نامه انگلیسی به فارسی ۹۶

واژه نامه فارسی به انگلیسی ۱۰۰

فهرست تصاویر

۵۷	 روابط بین ۱۲ کلاس	۱.۳
۵۹	 گراف $\Psi(5, p)$	۲.۳

فهرست جداول

۲۵		کران‌های بالا و پایین برای d_n	۱.۲
۳۷		بازی کلاه سه‌نفره متقارن	۱.۳
۶۱		بازی کلاه پنج‌نفره نامتقارن	۲.۳

فهرست اختصارات

C

CMDD Complete Minimum Distance Decoding
CMLD Complete Maximum Likelihood Decoding

D

DMS..... Discrete Memoryless Channel

I

IMDD Incomplete Minimum Distance Decoding
IMLD..... Incomplete Maximum Likelihood Decoding

M

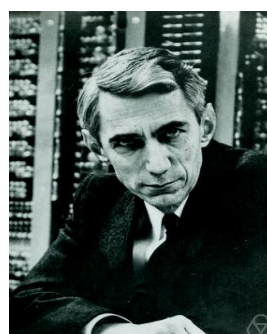
MAS..... Multi Agent Systems
MDD Minimum Distance Decoding
MLD Maximum Likelihood Decoding

فصل ۱

مفاهیم و تعاریف مقدماتی

۱.۱ مفاهیم مقدماتی در نظریه کدگذاری

تصویر کلاود شانون (۱۰ اردیبهشت ۱۲۹۵ خورشیدی، ۶ اسفند ۱۳۹۷)؛ ریاضی‌دان، مهندس الکترونیک و رمزنگار معروف آمریکایی، اهل میشیگان است که به‌عنوان پدر نظریه اطلاعات شناخته می‌شود.



نظریه کدگذاری^۱ در زمره کاربردی‌ترین شاخه‌های ریاضیات قرار دارد. این مبحث علاوه بر کاربردهای گسترده‌ای که در زمینه مهندسی برق و کامپیوتر دارد، در بخش‌های متنوعی از ریاضیات و نظریه علوم کامپیوتر نیز به‌طور طبیعی ظاهر شده و به‌عنوان ابزاری سودمند در این شاخه‌ها مورد استفاده قرار می‌گیرد. به‌عنوان نمونه، می‌توان از کاربرد نظریه کدگذاری در مباحث مولدهای شبه‌تصادفی، فشردن سازی متن در نظریه علوم کامپیوتر و طرح‌های بلوکی در ترکیبیات نام برد.

از سوی دیگر، ارتباط نظریه کدگذاری با شاخه‌های مختلف ریاضی موجب شده است تا روش‌های متنوعی از بخش‌های گوناگون ریاضی برای حل مسائل آن استفاده شود. از جمله این روش‌ها می‌توان به رویکردهای جبری، ترکیبیاتی، احتمالاتی، هندسه جبری، نظریه اعداد و نظریه گراف اشاره کرد. این موضوع سبب شده است تا مسائل نظریه کدگذاری برای طیف وسیعی از محققان شاخه‌های مختلف ریاضیات و علوم کامپیوتر جذابیت داشته باشد و محققانی با تخصص‌های مختلف در این زمینه مشغول به پژوهش باشند.

¹Coding theory

شروع کلاسیک نظریه کدگذاری به سال ۱۹۴۸ و انتشار مقاله کلاود شانون^۱ برمی گردد. در ابتدا این نظریه در حوزه نظامی و امنیتی گسترش یافت؛ اما بعد از توسعه فناوری اینترنت و مخابرات در دیگر حوزه‌ها نیز وارد شد. بحث کدگذاری اغلب در مورد چگونگی تصحیح خطای کدی است که فرستنده به گیرنده ارسال کرده است.

برای نخستین بار کلاود شانون در سال ۱۹۴۸، مقاله نظریه ریاضی ارتباطات را منتشر کرد که در آن نشان داد برای یک کانال ارتباطی پارازیت‌دار، عددی به عنوان ظرفیت کانال وجود دارد که در صورت به کارگیری روش‌های مناسب کدگذاری و کدگشایی، ارتباط معتبر با سرعت کمتر از ظرفیت کانال قابل دستیابی است. نظریه کدگذاری در حدود بیش از نیم قرن، رشدی شگفت‌انگیز پیدا کرده است. این نظریه در بیشتر حوزه‌ها، از سیستم‌های ارتباطی و پخش‌کننده‌های لوح فشرده گرفته تا فناوری ذخیره‌سازی بسیار کاربرد داشته است.

نظریه کدگذاری به بررسی روش‌های کدگذاری اطلاعات می‌پردازد و یکی از موضوع‌های مهم در بخش‌های مختلف علوم (مثل نظریه اطلاعات، ریاضیات و علوم رایانه، مهندسی برق و انتقال داده) است. در حالت کلی، واژه کدگذاری در سه زمینه مختلف به کاررفته است:

۱. فشرده‌سازی اطلاعات^۲: در ارتباط با کدگذاری منبع اطلاعاتی می‌باشد؛ به گونه‌ای که تا حد امکان فضا و حجم کمتری را اشغال کند. این امر با حذف افزونگی‌ها صورت می‌پذیرد. کدگذاری منبع اطلاعاتی، بخشی از نظریه اطلاع می‌باشد.

۲. کد تصحیح‌کننده خطا^۳: در ارتباط با افزایش قابلیت اطمینان در سیستم‌های ارتباطی پارازیت‌دار می‌باشد و با اضافه کردن بخش یا بخش‌هایی برای تشخیص و تصحیح خطا صورت می‌پذیرد. برخلاف قسمت قبل، این عمل با اضافه کردن بیت‌هایی به پیام اصلی اتفاق می‌افتد. این اضافه کردن در قسمت کانال رخ می‌دهد.

۳. رمزنگاری^۴: در ارتباط با امنیت و خصوصی‌سازی اطلاعات می‌باشد، هدف این است که اطلاعات ارسال شده به غیر از هدف اصلی (گیرنده پیام)، برای افراد دیگر قابل فهم و تشخیص نباشد.

تعریف ۱.۱.۱. فرض کنید A یک مجموعه q -تایی به صورت $A = \{a_1, a_2, \dots, a_q\}$ باشد. مجموعه A الفبای کد^۵ نامیده می‌شود و هر یک از عناصر مجموعه A را سمبل (نماد) کد^۶ گویند. همچنین فرض کنید A^n تمامی رشته‌های به طول n روی مجموعه A باشد؛ یعنی:

$$A^n = \{x_1 x_2 \dots x_n \mid x_i \in A\}$$

در این صورت:

¹Claude Shannon

²Data compression

³Error correcting code

⁴Cryptology

⁵Code alphabet

⁶Code symbol

الف) هر زیرمجموعهٔ ناتهی مانند C از A^n یک کد بلوکی^۱ q -تایی نامیده می‌شود.

ب) هر عضو C یک کدواژه^۲ نامیده می‌شود.

ج) تعداد کدواژه‌های کد C را $|C|$ اندازۀ کد نامیده و با نماد $|C|$ نمایش داده می‌شود.

د) یک کد از طول n و اندازۀ M را یک (n, M) -کد گویند.

گاهی اوقات، کد بلوکی q -تایی C را کد q -تایی یا به‌طور خلاصه‌تر کد می‌گویند.

ملاحظه ۱.۱.۱. اکثر مواقع الفبای کد یک میدان متناهی در نظر گرفته می‌شود؛ در این صورت $A = F_q$ می‌باشد.

• اگر $A = F_2$ باشد؛ در این صورت کد را یک کد دودویی^۳ می‌نامند.

• اگر $A = F_3$ باشد؛ در این صورت کد را یک کد سه‌تایی^۴ می‌نامند.

• اگر $A = F_4$ باشد؛ در این صورت کد را یک کد چهارتایی^۵ می‌نامند.

مثال ۱.۱.۱. $C_1 = \{0011, 0101, 1010, 1100, 1001, 0110\}$ یک $(4, 6)$ -کد دودویی می‌باشد؛ زیرا از ۶ کدواژه تشکیل شده که هر کدواژه دارای طول ۴ می‌باشد و الفبای کد میدان دو عنصری $A = F_2 = \{0, 1\}$ است.

$C_2 = \{01201, 12200, 21011\}$ یک $(5, 3)$ -کد سه‌تایی با الفبای کد $F_3 = \{0, 1, 2\}$ می‌باشد.

تعریف ۲.۱.۱. نرخ اطلاعات^۶ یک کد به‌صورت $R = \frac{k}{n}$ می‌باشد، جایی که k تعداد بیت‌های اطلاعات مفید و n تعداد بیت‌هایی است که کدکننده صرف کدکردن اطلاعات و داده‌ها می‌کند؛ یعنی همیشه $n \geq k$ است و به تعداد $n - k$ بیت اضافی در فرآیند کدگذاری اعمال می‌شود. در واقع نرخ هر کد، نسبت تعداد بیت‌های پیام ارسالی به تعداد بیت‌های پیام کدشده می‌باشد. برای هر کد بلوکی q -تایی C ، نرخ کد به‌صورت زیر تعریف می‌شود:

$$R = \frac{\log_q |C|}{n}.$$

نرخ کد کمیتی برای اندازه‌گیری کارایی کد می‌باشد و کدهای با نرخ بالا از اهمیت ویژه‌ای در نظریهٔ کدگذاری برخوردار می‌باشد. هرچند در عمل به‌غیر از فاکتور نرخ کد، قابلیت تشخیص و تصحیح خطای کد نیز از اهمیت ویژه‌ای برخوردار می‌باشد. بالاترین نرخ کد برابر با $R = 1$ می‌باشد که اگر یک کد دارای نرخ ۱ باشد، قابلیت تشخیص و تصحیح هیچ خطایی را ندارد.

¹Block code

²Codeword

³Binary code

⁴Ternary code

⁵Quaternary code

⁶Information rate

کانال: یک کانال مخابراتی را می‌توان به این صورت فرض کرد که یک کدواژه $c = c_1 \dots c_n$ از کد C (با طول n روی الفبای A) را دریافت نموده و خروجی آن یک رشته خروجی $d = d_1 \dots d_n$ (با همان طول n روی الفبای A) است.

تعریف ۳.۱.۱. کانال بدون حافظه^۱ گسسته شامل یک الفبای ورودی $F_q = A = \{a_1, \dots, a_q\}$ و یک الفبای خروجی $O = \{b_1, \dots, b_t\}$ ($A \subseteq O$) و یک مجموعه از احتمالات کانال یا احتمال‌های انتقال $P(b_j|a_i)$ است که در رابطه زیر صدق می‌کنند:

$$\forall i, \sum_{j=1}^t P(b_j|a_i) = 1;$$

که $P(b_j|a_i)$ احتمال آن است که b_j دریافت شده باشد به شرط آن که a_i ارسال شده باشد. علاوه بر این، اگر $c = c_1, \dots, c_n$ و $d = d_1, \dots, d_n$ به ترتیب کلماتی به طول n روی A و O باشند؛ آن‌گاه $P(d|c) = \prod_{i=1}^n P(d_i|c_i)$.

مثال ۲.۱.۱. مهم‌ترین کانال بدون حافظه گسسته (DMS)، کانال دودویی متقارن^۲ است که به صورت زیر می‌باشد:

$$P(1|0) = P(0|1) = P, \quad P(0|0) = P(1|1) = 1 - P.$$

بنابراین احتمال یک خطای تک‌بیتی یا احتمال متقاطع^۳ برابر با P است.

تعریف ۴.۱.۱. فرض کنید a و b دو کلمه به طول n روی الفبای A باشد، فاصله همینگ^۴ بین دو کدواژه $a = (a_1, \dots, a_n)$ و $b = (b_1, \dots, b_n)$ که با $d(a, b)$ نمایش داده می‌شود، به صورت تعداد جایگاه‌های متفاوت a و b تعریف می‌شود؛ یعنی تعداد بیت‌هایی که a و b با هم تفاوت دارند. لذا داریم:

$$d_H(a, b) = |\{i \mid a_i \neq b_i; i = 1, \dots, n\}|.$$

به بیان معادل $d_H(a, b) = \sum_{i=1}^n d_H(a_i, b_i)$ می‌باشد؛ جایی که،

$$d_H(a_i, b_i) = \begin{cases} 1; & a_i \neq b_i \\ 0; & a_i = b_i. \end{cases}$$

تعریف ۵.۱.۱. فرض کنید C یک کد بلوکی دارای حداقل دو کدواژه باشد؛ در این صورت فاصله کد^۵ C که با نماد $d(C)$ نمایش داده می‌شود، به صورت مینیمم فاصله همینگ بین تمام کدواژه‌های کد C تعریف می‌شود، داریم:

$$d(C) = \min\{d_H(a, b) \mid a, b \in C; a \neq b\}.$$

¹Discrete memoryless channel

²Binary symmetric channel

³Crossover probability

⁴Hamming distance

⁵Distance of a code

ملاحظه ۲.۱.۱. هر کد بلوکی که طول آن n (طول کدواژه‌های کد بلوکی)، اندازه آن M (تعداد کدواژه‌های کد C) و فاصله همینگ آن d باشد را یک (n, M, d) - کد می‌نامند؛ در این صورت اعداد n, M و d پارامترهای کد می‌باشند.

تعریف ۶.۱.۱. مجموعه ناتهی V به همراه دو عمل «+» و «.» را فضای برداری روی میدان F_q گویند؛ هرگاه برای $u, v, w \in V$ و $\lambda, \mu \in F_q$ داشته باشیم:

$$1. u + v \in V.$$

$$2. (u + v) + w = u + (v + w).$$

$$3. \text{عنصری مانند } \circ \in V \text{ وجود دارد که برای هر } v \in V \text{ داریم: } \circ + v = v = v + \circ.$$

$$4. \text{برای هر } u \in V, \text{ عنصری از } V \text{ موجود است به قسمی که } u + (-u) = (-u) + u = \circ.$$

$$5. u + v = v + u.$$

$$6. \text{برای هر } v \in V \text{ و } \lambda \in F_q, \lambda.v \in V.$$

$$7. \text{برای هر } v, w \in V, \lambda \in F_q \text{ داریم } \lambda.(v + w) = \lambda.v + \lambda.w.$$

$$8. \text{برای هر } v \in V, \lambda, \mu \in F_q \text{ داریم } (\lambda\mu).v = \lambda.(\mu v).$$

$$9. \text{اگر عنصر } 1 \text{ همانی ضربی } F_q \text{ باشد؛ آن گاه } 1_F.v = v \text{ برای } \forall v \in V.$$

تعریف ۷.۱.۱. فرض کنید V یک فضای برداری روی میدان متناهی F_q و $S = \{v_1, v_2, \dots, v_n\}$ زیرمجموعه ناتهی از فضای برداری V باشد؛ در این صورت:
الف) S یک زیرفضای برداری از V می‌باشد اگر و تنها اگر:

$$\forall v_1, v_2 \in S, \forall \lambda, \mu \in F_q; \lambda v_1 + \mu v_2 \in S.$$

ب) S یک مجموعه مستقل خطی است؛ هرگاه برای $\lambda_1, \dots, \lambda_n \in F_q$ داشته باشیم:

$$\lambda_1 v_1 + \dots + \lambda_n v_n = \circ \implies \lambda_1 = \lambda_2 = \dots = \lambda_n = \circ.$$

لازم به ذکر است که اگر مجموعه S مستقل خطی نباشد، آن را وابسته خطی گویند.

ج) S را یک مولد برای فضای برداری V نامند؛ هرگاه داشته باشیم:

$$V = \langle S \rangle = \{\lambda_1 v_1 + \dots + \lambda_n v_n \mid \lambda_i \in F_q, v_i \in S, 1 \leq i \leq n\}.$$

د) S یک پایه برای V است؛ هرگاه:

– S مستقل خطی باشد.

– S مولد باشد.

نکته: تعداد عناصر پایه فضای برداری V روی میدان متناهی F_q را بُعد فضای برداری نامیده و با نماد $\dim_{F_q}(V)$ نمایش داده می‌شود.

تعریف ۸.۱.۱. هر زیرفضایی از فضای برداری F_q^n ، یک کد خطی^۱ از طول n روی میدان متناهی F_q نامیده می‌شود. در این حالت اگر C دارای بُعد k باشد، آن را به صورت $[n, k]$ – کد نمایش می‌دهند و اگر $d(C) = d$ فاصله کد C باشد؛ آن گاه C را یک $[n, k, d]$ – کد گویند. لازم به ذکر است که اگر کد C خطی نباشد، آن را غیرخطی می‌نامند. از آنجا که کدهای خطی فضاهای برداری هستند، ساختار جبری آن‌ها موجب می‌شود توصیف و استفاده آن‌ها نسبت به کدهای غیرخطی راحت‌تر باشد.

تعریف ۹.۱.۱. فرض کنید $x = x_1 \dots x_n$ یک کلمه در F_q^n باشد؛ در این صورت وزن همینگ کلمه x که با نماد $wt(x)$ نمایش داده می‌شود، برابر با تعداد مؤلفه‌های ناصفر کلمه x تعریف می‌شود؛ لذا داریم: $wt(x) = d(x, \circ)$ ، جایی که $\circ = (\circ, \circ, \dots, \circ) \in F_q^n$. به عبارت دیگر، $wt(x) = \sum_{i=1}^n wt(x_i)$ است جایی که،

$$wt(x_i) = d(x_i, \circ) = \begin{cases} 1 & ; x_i \neq \circ \\ \circ & ; x_i = \circ. \end{cases}$$

تعریف ۱۰.۱.۱. فرض کنید C یک کد خطی در F_q^n باشد، دوگان کد C که با نماد $C^\perp$ نمایش داده می‌شود به صورت زیر تعریف می‌شود:

$$C^\perp = \{v \in F_q^n \mid v.c = \circ, \forall c \in C\}.$$

تعریف ۱۱.۱.۱. ماتریسی که سطرهای آن پایه‌ای برای کد خطی C باشد ماتریس مولد^۲ و ماتریسی که سطرهای آن پایه‌ای برای دوگان کد C باشد ماتریس کنترل توازن^۳ کد خطی C نامیده می‌شود.

^۱Linear code

^۲Generator matrix

^۳Parity-check matrix

۲.۱ قاعده‌های کدگشایی

۱.۲.۱ قاعده کدگشایی ماکزیمم احتمال (MLD)

فرض کنید کدواژه‌های کد C از طریق یک کانال پارازیت‌دار ارسال شده باشند و کلمه x را دریافت کرده باشیم؛ در این صورت قاعده کدگشایی ماکزیمم احتمال^۱، کلمه x را به کدواژه c_x کدگشایی می‌کند هرگاه احتمال ارسال برای c_x به شرط دریافت x ماکزیمم باشد؛ یعنی داشته باشیم:

$$P(x \text{ دریافت} | c_x \text{ ارسال}) = \max \{ P(x \text{ دریافت} | c) | c \in C \}.$$

دو نوع قاعده کدگشایی ماکزیمم احتمال داریم:

۱. قاعده کدگشایی ماکزیمم احتمال کامل (CMLD)

در این روش اگر کلمه x دریافت شود و تعداد کدواژه‌هایی که به شرط دریافت کلمه x دارای ماکزیمم احتمال ارسال است، بیش از یک مورد باشد؛ آن‌گاه یکی از کدواژه‌ها را به دلخواه انتخاب کرده و کلمه x را به آن کدواژه کدگشایی می‌کند.

۲. قاعده کدگشایی ماکزیمم احتمال غیرکامل (IMLD)

در این روش اگر کلمه x دریافت شود و تعداد کدواژه‌هایی که به شرط دریافت کلمه x دارای ماکزیمم احتمال ارسال است، بیش از یک مورد باشد؛ آن‌گاه بدون هیچ‌گونه تصمیم‌گیری صرفاً درخواست ارسال مجدد کد را انجام می‌دهد.

۲.۲.۱ قاعده کدگشایی مینیمم فاصله (MDD)

فرض کنید کدواژه‌های کد C در یک کانال ارتباطی پارازیت‌دار ارسال شده باشند و کلمه x را دریافت کرده باشیم؛ در این صورت قاعده کدگشایی مینیمم فاصله^۲، کلمه x را به کدواژه c_x کدگشایی می‌کند؛ هرگاه فاصله بین x و c_x در بین تمام کدواژه‌های کد C ، مینیمم مقدار ممکن باشد؛ یعنی داشته باشیم:

$$d_H(x, c_x) = \min \{ d_H(x, c) | c \in C \}.$$

دو نوع قاعده کدگشایی مینیمم فاصله داریم:

۱. قاعده کدگشایی مینیمم فاصله کامل (CMDD)

در این روش اگر کلمه x دریافت شود و تعداد کدواژه‌هایی که با کلمه x دارای مینیمم

^۱Maximum likelihood decoding

^۲Minimum distance decoding

فاصله است، بیش از یک مورد باشد؛ آن گاه یکی از کدواژه‌ها را به دلخواه انتخاب کرده و کلمه x را به آن کدواژه کدگشایی می‌کند.

۲. قاعده کدگشایی مینیم فاصله غیرکامل (IMDD)

در این روش اگر کلمه x دریافت شود و تعداد کدواژه‌هایی که با کلمه x دارای مینیم فاصله است، بیش از یک مورد باشد؛ آن گاه بدون هیچ گونه تصمیم‌گیری صرفاً درخواست ارسال مجدد کد را انجام می‌دهد.

تعریف ۱.۲.۱. کد C تشخیص‌دهنده t خطا^۱ نامیده می‌شود، اگر حداقل یک خطا و حداکثر t خطا روی هر کدواژه رخ دهد؛ آن گاه کلمه دریافتی، یک کدواژه از کد C نباشد. همچنین کد C یک کد دقیقاً تشخیص‌دهنده t خطا نامیده می‌شود؛ هرگاه تشخیص‌دهنده t خطا باشد؛ اما تشخیص‌دهنده $t+1$ خطا نباشد.

بدیهی است که کد C ، یک کد تشخیص‌دهنده t خطاست اگر و تنها اگر $d(C) \geq t+1$ باشد. و هر کد C با فاصله $d(C) = d$ ، یک کد دقیقاً $d-1$ تشخیص‌دهنده خطا می‌باشد.

تعریف ۲.۲.۱. کد C تصحیح‌کننده t خطا^۲ نامیده می‌شود، اگر قاعده MDD قادر به تصحیح حداکثر t خطا در هر کدواژه باشد. همچنین کد C تصحیح‌کننده دقیقاً t خطاست اگر تصحیح‌کننده t خطا باشد؛ اما تصحیح‌کننده $t+1$ خطا نمی‌باشد.

می‌توان دید که کد C ، یک کد تصحیح‌کننده t خطاست اگر و تنها اگر $d(C) \geq 2t+1$ باشد. و فرض کنید کد C یک کد با فاصله $d(C) = d$ باشد؛ در این صورت کد C دقیقاً $\lfloor \frac{d-1}{2} \rfloor$ تصحیح‌کننده خطا می‌باشد.

تعریف ۳.۲.۱. کد دودویی C که $d(C) \geq 4$ است، یک کد دودویی ۱-تصحیح‌کننده خطا و ۲-تشخیص‌دهنده خطا می‌باشد، این کد را به اختصار ۱-تصحیح و ۲-تشخیص‌کد گوئیم. و به صورت «کد SECDED» نمایش داده می‌شود.

تعریف ۴.۲.۱. $C \subseteq \{0,1\}^n$ را یک کد پوششی دودویی^۳ از شعاع R و طول n گوئیم؛ هرگاه برای هر $x \in \{0,1\}^n$ حداقل یک $y \in C$ وجود داشته باشد به طوری که $d(x,y) \leq R$.

۳.۱ کران‌هایی در نظریه کدگذاری

فرض کنید یک (n, M, d) - کد q -تایی موجود باشد؛ در این صورت کارایی کد با مقدار M و d در ارتباط است. یک (n, M, d) - کد خوب باید دارای دو شرط زیر باشد:

¹T-error detecting

²T-error correcting

³Binary covering code

الف) دارای اندازه (M) بزرگ باشد تا بتواند تعداد بیشتری از پیامهای منبع را کد نماید.

ب) دارای کمترین فاصله (d) ممکن باشد تا بتواند خطاهای بیشتری را تصحیح کند.

تعریف ۱.۳.۱. فرض کنید مجموعه الفبای A مفروض باشد به طوری که $|A| = q \geq 1$ ، و مقادیر n و d مشخص باشند. در این صورت عدد $A_q(n, d)$ را ماکزیمم اندازه ممکن برای M ؛ به گونه ای که یک (n, M, d) - کد موجود باشد، تعریف می شود. به عبارت دیگر،

$$A_q(n, d) = \max\{M \mid (n, M, d) \text{ - کد موجود باشد}\}.$$

یک (n, M, d) - کد که در آن $A_q(n, d) = M$ باشد، کد بهینه^۱ نامیده می شود.

ملاحظه ۱.۳.۱. محاسبه مقدار $A_q(n, d)$ مسأله ای جالب و تأثیرگذار در نظریه کد می باشد که از آن به عنوان مهم ترین مسئله نظریه کدگذاری یاد می شود. هرچند محاسبه مقدار دقیق $A_q(n, d)$ کار ساده ای نمی باشد؛ ولی می توان برای آن کرانهایی بالا و پایین مشخص کرد که در عمل، کار را راحت می کند.

۱.۳.۱ کرانه های پایین برای کدها

تعریف ۲.۳.۱. فرض کنید A یک الفبای q - عنصری باشد ($|A| = q \geq 1$). در این صورت برای هر بردار $u \in A^n = \{u = (u_1, \dots, u_n) \mid u_i \in A\}$ و هر عدد صحیح $r \geq 0$ ، گوی n - بُعدی به شعاع r و مرکز u را با نماد $S_A(u, r)$ نمایش داده می شود و به صورت زیر تعریف می شود:

$$S_A(u, r) = \{v \in A^n \mid d(u, v) \leq r\}.$$

تعریف ۳.۳.۱. فرض کنید $q > 1$ و $r \geq 0$ اعدادی صحیح و n عددی صحیح مثبت باشد؛ در این صورت عدد $V_q^n(r)$ به صورت زیر تعریف می شود:

$$V_q^n(r) = \begin{cases} \sum_{m=0}^r \binom{n}{m} (q-1)^m; & 0 \leq r \leq n \\ q^n; & r \geq n. \end{cases}$$

برای هر عدد صحیح $r \geq 0$ ، گوی n - بُعدی با شعاع r در A^n دارای دقیقاً $V_q^n(r)$ بردار می باشد، جایی که A الفبای q - عنصری باشد.

کران گوی پوششی: برای هر عدد صحیح $q > 1$ و اعداد صحیح n و d به گونه ای $1 \leq d \leq n$ باشد، همواره داریم:

$$\frac{q^n}{\sum_{i=0}^{d-1} \binom{n}{i} (q-1)^i} \leq A_q(n, d).$$

¹Optimal code

۲.۳.۱ کران‌های بالا برای کدها

کران زیر معروف به کران همینگ را بیان می‌کنیم که در حالت مرزی منجر به ایجاد رده‌ای مهم از کدها به نام کدهای کامل می‌شود.

کران همینگ: فرض کنید $q > 1$ عددی صحیح و n و d اعدادی صحیح باشند به گونه‌ای که $2 \leq d \leq n$ باشد؛ آن‌گاه همواره داریم:

$$A_q(n, d) \leq \frac{q^n}{\sum_{i=0}^{\lfloor \frac{d-1}{q} \rfloor} \binom{n}{i} (q-1)^i}.$$

تعریف ۴.۳.۱. یک کد q -تایی که نامساوی کران همینگ را به مساوی تبدیل می‌کند، کد کامل^۱ نامیده می‌شود. در واقع یک کد کامل دارای $\frac{q^n}{\sum_{i=0}^{\lfloor \frac{d-1}{q} \rfloor} \binom{n}{i} (q-1)^i}$ کدواژه می‌باشد.

تعریف ۵.۳.۱. فرض کنید $r \geq 2$ باشد. یک کد خطی دودویی از طول $n = 2^r - 1$ ، با ماتریس کنترل توازن H که ستون‌های آن شامل تمام بردارهای ناصفر از F_2^r می‌باشد را کد همینگ دودویی^۲ از طول $2^r - 1$ نامند و با نماد $Ham(r, 2)$ یا $H_2(r)$ نشان داده می‌شود. کدهای همینگ شاید معروف‌ترین کدهای تصحیح‌کننده خطا باشند. این کدها، به‌طور مستقل توسط گلی^۳ در سال ۱۹۴۹ و همینگ^۴ در سال ۱۹۵۰ کشف شدند. کدهای همینگ خطی و کامل بوده و کدگشایی بسیار ساده‌ای دارند.

خواص کدهای همینگ دودویی:

- بُعد $Ham(r, 2)$ برابر $k = 2^r - 1 - r$ است.
- فاصله $Ham(r, 2)$ برابر $d = 3$ می‌باشد؛ بنابراین کد همینگ دودویی یک کد دقیقاً یک‌تصحیح‌کننده خطا و همچنین یک کد دوتشخیص‌دهنده خطا است.

¹Perfect code

²Binary Hamming code

³Marcel Golay

⁴Richard Hamming

۴.۱ مفاهیم مقدماتی در نظریه رمزنگاری

رمزنگاری پیشینه طولانی و درخشان دارد که به هزاران سال قبل برمی گردد. امروزه رمزنگاری در قلب ارتباطات مبتنی بر اینترنت، تجارت الکترونیک، پرداخت‌های بانکی و محصولات مبتنی بر فناوری زنجیره بلوک مانند بیت کوین^۱ قرار دارد. کلمه «Cryptography» برگرفته از لغات یونانی «Kryptos» به معنای «محرمانه» و «graphic» به معنای «نوشتن» است؛ بنابراین کلمه رمزنگاری به معنای محرمانه‌نوشتن می‌باشد. رمزنگاری دانشی است که به بررسی و شناخت اصول و روش‌های انتقال یا ذخیره اطلاعات به صورت امن (حتی اگر مسیر انتقال اطلاعات و کانال‌های ارتباطی یا محل ذخیره اطلاعات ناامن باشند) می‌پردازد. رمزنگاری استفاده از تکنیک‌های ریاضی، برای برقراری امنیت اطلاعات است.



در اصل رمزنگاری دانش تغییر دادن متن پیام یا اطلاعات به کمک کلید رمز و با استفاده از یک الگوریتم رمز است، به صورتی که تنها شخصی که از کلید و الگوریتم مطلع است قادر به استخراج اطلاعات اصلی از اطلاعات رمز شده باشد و شخصی که از یکی یا هر دوی آن‌ها اطلاع ندارد، نتواند به اطلاعات دسترسی پیدا کند. دانش رمزنگاری بر پایه مقدمات

بسیاری از قبیل تئوری اطلاعات، نظریه اعداد و آمار بنا شده است و امروزه به طور خاص در علم مخابرات مورد بررسی و استفاده قرار می‌گیرد. در رمزنگاری محتویات یک متن به صورت حرف به حرف و در بعضی موارد بیت به بیت تغییر داده می‌شود و هدف، تغییر محتوای متن است نه تغییر ساختار زبان شناختی آن. در مقابل، کدگذاری تبدیلی است که کلمه‌ای را با یک کلمه یا نماد دیگر جایگزین می‌کند و ساختار زبان شناختی متن را تغییر می‌دهد.

تعریف ۱.۴.۱. اتحاد موقت گروه‌ها که هدف‌های معینی را دنبال می‌کنند، ائتلاف^۲ گویند.

تعریف ۲.۴.۱. سیستم مجموعه‌ای از اجزا است که برای رسیدن به یک هدف مشترک با همدیگر تعامل دارند. به عبارت دیگر، سیستم مجموعه‌ای از فرایندها است که در ارتباط با یکدیگر هستند و برای رسیدن به هدفی مشخص تلاش می‌کنند.

تعریف ۳.۴.۱. سیستم چندعاملی^۳ به سیستمی گفته می‌شود که از تعدادی عامل تشکیل شده است. این عامل‌ها هر کدام به نوبه خود فعل و انفعالات داخلی داشته و در محیط خارج نیز با یکدیگر ارتباط دارند.

¹Bitcoin

²Coalition

³Multi agent systems

سیستم چندعاملی یا به اختصار MAS، روش نوینی برای حل مسائل و پیاده‌سازی پروژه‌های نرم‌افزاری رایانه‌ای است.

تعریف ۴.۴.۱. مسائل تصمیم^۱ به مسائلی گفته می‌شود که هدف در آن‌ها بررسی درستی یا نادرستی یک ورودی است؛ به عبارت دیگر، خروجی همواره درست (True) یا غلط (False) می‌باشد.

تعریف ۵.۴.۱. به تابع $f: \{0, 1\}^* \rightarrow \{0, 1\}$ یک تابع بولی^۲ گوییم که $\{0, 1\}^* = \bigcup_{n=1}^{\infty} \{0, 1\}^n$ است. جواب عبارات بولی همواره یا درست (یک) یا غلط (صفر) می‌باشد. یک تابع بولی مشخص می‌کند که چگونه به وسیله محاسبات منطقی و ورودی‌های بولی، خروجی مناسب بسازیم. چنین توابعی نقش اساسی در مسائل نظریه پیچیدگی دارند. توابع بولی نقش بسیار مهمی را در رمزنگاری، به‌ویژه در طراحی الگوریتم‌های کلید متقارن ایفا می‌کنند.

تعریف ۶.۴.۱. تابع آستانه‌ای^۳ یک تابع بولی است که به صورت زیر تعریف می‌شود:

$$f(x_1, \dots, x_n) = \begin{cases} 1; & w_1x_1 + w_2x_2 + \dots + w_nx_n \geq t \\ 0; & o.w. \end{cases}$$

که در آن عدد حقیقی t آستانه تابع و $w_1, w_2, \dots, w_n$ اعداد حقیقی از پیش تعیین شده‌ای هستند که وزن‌ها نامیده می‌شوند.

لم ۱.۴.۱ (لم اسپرنر^۴). اگر برای خانواده^۵ $\{A_1, \dots, A_m\} \subseteq 2^{[n]}$ داشته باشیم: $A_i \not\subseteq A_j$ برای هر $1 \leq i \neq j \leq m$ ، آن‌گاه $m \leq \binom{n}{\lfloor \frac{n}{2} \rfloor}$ می‌باشد.

تعریف ۷.۴.۱. یک ابرگراف^۵ $H = (V, E)$ شامل یک مجموعه غیرتهی از رأس‌ها به صورت $V = \{v_1, \dots, v_n\}$ و یک خانواده^۵ E ، از زیرمجموعه‌هایی از رئوس به صورت $E = \{e_1, \dots, e_m\}$ است. مجموعه‌های E_j ، یال‌های ابرگراف یا ابریال می‌باشند. حالت $K = 2$ را به اختصار گراف گوییم.

تعریف ۸.۴.۱. گراف $G = (V, E)$ را دوبخشی نامیم اگر دارای دو شرط زیر باشد:

(الف) مجموعه رئوس G را بتوان به صورت اجتماع دو زیرمجموعه غیرتهی به صورت $V_1 \cup V_2 = V$ در نظر گرفت.

(ب) دو انتهای هیچ یالی از G در یک بخش واقع نباشد.

¹Decision problem

²Boolean function

³Threshold function

⁴Sperner's lemma

⁵Hyper graph

گراف $G = (V, E)$ را K -بخشی نامیم اگر مجموعه رؤوس آن به K زیرمجموعهٔ غیرتهی طوری افراز گردد به طوری که دو سر هیچ یالی در یک بخش واقع نشود.

تعریف ۹.۴.۱. مجموعهٔ $D \subseteq V$ از گراف $G = (V, E)$ احاطه‌گر^۱ نامیده می‌شود اگر هر رأس $V \setminus D$ حداقل یک همسایه در D داشته باشد. مسئلهٔ مجموعهٔ احاطه‌گر مینیمم، مجموعهٔ احاطه‌گر با اندازهٔ مینیمم را جست‌وجو می‌کند. به عبارت دیگر، مجموعهٔ D مجموعهٔ احاطه‌گر است، اگر:

$$\forall v \in V \quad \exists x \in D \quad \text{s.t.} \quad vx \in E.$$

^۱Dominating set

فصل ۲

کدهای پوششی و کران‌های بالا روی ابعاد بازی‌های ساده

۱.۲ معرفی

یک سیستم چندعامله را در نظر بگیرید که در آن یک ائتلاف، بین عوامل به‌منظور حل یک مسئله داده‌شده، شکل گرفته است و ما باید پیش‌بینی کنیم که آیا ائتلاف موفق خواهد بود یا خیر. توجه خود را معطوف به حالت‌هایی کرده‌ایم که از حالت طبیعی یکنوا پیروی می‌کند؛ این حالت گویای این است که ابرمجموعه از هر ائتلاف موفق، خود نیز موفق خواهد بود. در چنین سیستم چندعامله‌ای، به یک سیستم برای محاسبه پیش‌بینی «آره» یا «نه» نیاز داریم. بازی‌های ساده می‌توانند به‌صورت یک تابع بولی یکنوا و یا ابرگراف یکنوا دیده شوند؛ لذا بازی‌های ساده برای مدل‌سازی سیستم‌هایی استفاده می‌شوند که تصمیم می‌گیرند آیا ائتلاف موفق (پیروز) بوده یا خیر (باخت). این عوامل در یک بازی ساده که اشاره به بازیکنان و ائتلاف موفق و ناموفق دارد، می‌تواند به ائتلاف برد و باخت نیز اشاره داشته باشد.

بازی وزنی، یک نوع خاص از بازی ساده است که برای هر بازیکن، یک وزن مشخص شده است و یک ائتلاف، موفق است اگر و تنها اگر مجموع وزن تمام بازیکنان آن ائتلاف برابر یا بیشتر از معیار مشخص شده باشد. هر بازی ساده می‌تواند به‌صورت یک یا تقاطع چند بازی وزنی بیان شود. ابعاد یک بازی ساده، کمترین تعداد بازی وزنی مورد نیاز برای نمایش آن بازی ساده است. یک بازی ساده از بُعد K است اگر و تنها اگر به‌عنوان دقیقاً K بازی وزنی نمایش داده شود و با کمتر از K این کار امکان‌پذیر نباشد. به‌عنوان مثال، یک بازی ساده از بُعد ۱ است اگر و تنها اگر وزنی باشد. ابعاد، تأثیر مستقیمی بر فضای ذخیره مورد نیاز و کارایی چنین سیستمی دارد؛ لذا درصدد آن هستیم تا ابعاد بازی‌های ساده را تا حد امکان بهبود بخشیم.

در این فصل d_n به‌عنوان بُعد ماکزیممی که می‌توان برای یک بازی ساده با n بازیکن به

دست آورد، در نظر گرفته می‌شود. تیلور و زویکر [۲۹] نشان دادند که چگونه می‌توان یک بازی ساده را به صورت تقاطع حداکثر $\left(\lfloor \frac{n}{4} \rfloor\right)$ بازی وزنی نشان داد و در نتیجه کران بالای $\left(\frac{n}{4}\right)$ را برای d_n اثبات کردند. هدف اصلی در این فصل، بهبود کران بالای ارائه شده توسط تیلور و زویکر می‌باشد. برای این منظور دو کران بالا ارائه می‌شود که هر دو نشانگر نسخه‌هایی قوی‌تر از آنچه توسط تیلور و زویکر ارائه شده، می‌باشد.

اخیراً السن و همکارانش [۲۴] یک بهبود در کران پایین d_n با استفاده از تئوری کدهای تصحیح خطا ثابت کردند. در این فصل از یک روش کاملاً متفاوت و جدید بر اساس کدهای پوششی، به منظور دستیابی به کران بالا استفاده شده است. فاصله بین کران بالا و پایین با استفاده از این روش برای بعضی از n ها از عامل n به $\sqrt{n}$ کاهش پیدا کرده است. همچنین در حالت کلی، این عامل به $\ln n \sqrt{n}$ کاهشیده شده است. در نهایت با پیشنهاد کردن یک روش برای کاهش احتمالی فاصله ابعادی نتیجه‌گیری می‌شود.

۲.۲ فعالیت‌های مرتبط

تیلور و زویکر [۲۹] دنباله‌ای از بازی‌ها را با ابعاد حداقل $2^{\frac{n}{4}-1}$ برای $n = 2k$ با k های فرد ساخته‌اند. در نهایت ثابت شد [۲۴] که ابعاد بازی‌های ساده ارائه شده توسط تیلور و زویکر دقیقاً مقدار $2^{\frac{n}{4}-1}$ است. فریکس و پوینت [۱۳] نشان دادند که چگونه نوع دیگری از بازی‌های ساده با ابعاد $2^{\frac{n}{4}-1}$ را برای تمام n های زوج می‌توان ساخت. با توجه به تعریف d_n ، این نتایج نشان می‌دهد که $d_n \geq 2^{\frac{n}{4}-1}$ است. این کران پایین برای d_n ، اخیراً توسط السن و همکارانش [۲۴] با تشکیل ارتباطی بین کدهای تصحیح کننده خطا به صورت زیر بهبود یافته است:

$$d_n \geq \frac{1}{n} \binom{n}{\lfloor \frac{n}{4} \rfloor} \in 2^{n-o(n)}. \quad (1.2)$$

در نظر گرفتن معادله زیر برای مقایسه با مقدار کران پایین کمک کننده خواهد بود:

$$\binom{n}{\lfloor \frac{n}{4} \rfloor} = (1 - o(1)) \sqrt{\frac{2}{\pi n}} 2^n. \quad (2.2)$$

همچنین کورز و نیل [۲۱] یک روش کلی به منظور تعیین کران‌های پایین برای ابعاد یک بازی ساده معرفی کرده‌اند.

یک ائتلاف ماکسیمال باخت در یک بازی ساده، ائتلاف بازنده‌ای است که با اضافه کردن هر بازیکن خارج از این ائتلاف، تبدیل به یک ائتلاف برنده خواهد شد. فرض کنید L^M مجموعه ائتلاف‌های ماکسیمال باخت باشد. تیلور و زویکر [۲۹] روشی ارائه دادند که به کمک آن می‌توان هر بازی ساده را به صورت تقاطع حداکثر $|L^M|$ بازی وزنی نمایش داد. این نتیجه، کران بالای $|L^M|$ را برای d_n اثبات می‌کند. کورز و نیل [۲۱] الگوریتم‌های ابتکاری بر اساس

برنامه‌نویسی خطی عدد صحیح، برای نمایش یک بازی ساده داده‌شده به صورت تقاطع چند بازی وزنی، معرفی کردند.

ابعاد سیستم‌های رأی‌گیری دنیای واقعی مورد توجه چندین تحقیق قرار گرفته است. مؤسسه اصلاحات کانادا [۱۸] و سیستم قانون‌گذاری آمریکا [۲۸] ابعاد ۲ را دارند. سیستم‌های رأی‌گیری از شورای قانون‌گذاری هنگ‌کنگ [۶] و شورای اتحادیه اروپا [۱۱] دقیقاً ابعاد ۳ را دارند.

مشخصاً راه‌های دیگری برای نشان‌دادن بازی‌های ساده وجود دارد. مفهوم هم-بعد برای یک بازی ساده به عنوان کمترین تعداد بازی‌های وزنی لازم برای نمایش این بازی ساده به صورت اجتماع بازی‌های وزنی معرفی شده است [۱۲]. در نظر گرفتن ترکیبات دلخواه از اجتماع و اشتراک بازی‌های وزنی منجر به تعریف بعد بولی می‌شود [۱۰].

۳.۲ مقدمات

در این بخش، مفاهیم و تعاریفی که در این فصل مورد نیاز است معرفی شده است. در ابتدا تعریف دقیقی از بازی ساده ارائه می‌شود و سپس یک توضیح کوتاه راجع به کدهای پوششی را مرور خواهیم کرد.

۱.۳.۲ بازی‌های ساده

تعریف ۱.۳.۲. یک بازی ساده^۱ Γ ، زوج مرتب (N, W) است که در آن $N = \{1, \dots, n\}$ برای مقدار صحیح مثبت n و $W \subseteq 2^N$ مجموعه‌ای از زیرمجموعه‌های N است؛ به طوری که:

$$\emptyset \notin W \bullet$$

$$N \in W \bullet$$

$$S \subseteq T \subseteq N \text{ و } S \in W \implies T \in W \bullet$$

اعضای N به بازیکنان و زیرمجموعه‌های N به ائتلاف‌ها اشاره دارند. یک ائتلاف، برنده است اگر عضوی از W باشد؛ در غیر این صورت بازنده خواهد بود. مجموعه ائتلاف بازنده به شکل $L = 2^N \setminus W$ نشان داده می‌شود.

اولین شرط بیان می‌کند که ائتلاف بدون بازیکن بازنده است و شرط دوم تضمین می‌کند که ائتلاف شامل تمام بازیکنان، برنده می‌باشد. شرط سوم حالت یکنوایی است که در آن هر ابرمجموعه یک ائتلاف برنده نیز برنده است.

^۱Simple game

تعریف ۲.۳.۲. یک ائتلاف، ائتلاف ماکسیمال باخت^۱ است اگر بازنده باشد و هر ابرمجموعه سره از آن برنده باشد.

مجموعه تمام ائتلاف‌های ماکسیمال باخت را با نماد L^M نشان می‌دهیم.

تعریف ۳.۳.۲. ائتلاف مینیمال برنده^۲ ائتلافی است که خودش برنده و هر زیرمجموعه آن بازنده باشد.

مجموعه تمام ائتلاف‌های مینیمال برنده با نماد W^m نشان داده می‌شود.

بدیهی است که یک بازی ساده Γ می‌تواند با داشتن یکی از مجموعه ائتلاف‌های W, L, W^m یا L^M نیز به صورت یکتا تعیین شود.

سؤال ۱.۳.۲. آیا در این پایان‌نامه یک بازی ساده که به صورت یک بازی وزنی قابل نمایش نباشد، ارائه شده است؟

بازی‌های وزنی زیرمجموعه‌ای سره از بازی‌های ساده را شکل می‌دهند و به صورت زیر تعریف می‌شوند:

تعریف ۴.۳.۲. یک بازی ساده $\Gamma = (N, W)$ وزنی است اگر یک سهم $q \in \mathbb{R}_+$ و وزن‌های $w_1, w_2, \dots, w_n \in \mathbb{R}_+$ وجود داشته باشند؛ به طوری که $S \in W$ است اگر و تنها اگر $\sum_{i \in S} w_i \geq q$ باشد. در این شرایط از نماد $\Gamma = [q; w_1, w_2, \dots, w_n]$ استفاده می‌شود.

تقاطع $\Gamma_1 \cap \Gamma_2$ از بازی‌های $\Gamma_1(N, W_1)$ و $\Gamma_2(N, W_2)$ ، یک بازی ساده با بازیکنان N و $W = W_1 \cap W_2$ است. تیلور و زویکر [۲۹] نشان دادند که هر بازی ساده می‌تواند به صورت تقاطع $|L^M|$ بازی‌های وزنی Γ_T برای $T \in L^M$ نمایش داده شود. برای هر بازی $\Gamma = \bigcap_{T \in L^M} \Gamma_T$ می‌باشد. در بازی Γ_T ائتلاف S برنده است اگر و تنها اگر $S \not\subseteq T$. به سادگی می‌توان یک نمایش وزنی از Γ_T با استفاده از وزن‌های $^\circ$ و 1 به صورت زیر ارائه کرد:

بازی، دارای سهم 1 است و هر بازیکن در $N \setminus T$ دارای وزن 1 و بازیکنان دیگر دارای وزن $^\circ$ هستند.

تعریف ۵.۳.۲. بُعد^۳ یک بازی ساده Γ که با نماد $d(\Gamma)$ نشان داده می‌شود، کوچک‌ترین عدد صحیح مثبت d است؛ به طوری که $\Gamma = \bigcap_{i=1}^d \Gamma_i$ می‌باشد، جایی که بازی‌های $\Gamma_1, \dots, \Gamma_d$ وزنی هستند.

در این فصل d_n بزرگ‌ترین بُعد ممکن برای یک بازی ساده با n بازیکن تعریف می‌شود. به عبارت دیگر؛

$$d_n = \max\{d(\Gamma) \mid \Gamma \text{ یک بازی ساده با } n \text{ بازیکن است}\}.$$

¹Maximal losing coalition

²Minimal winning coalition

³Dimension

ملاحظه ۱.۳.۲. واضح است که یک ائتلاف ماکسیمال باخت نمی تواند شامل ائتلاف ماکسیمال باخت دیگری باشد.

بنابراین اگر Γ یک بازی ساده با n بازیکن باشد؛ آن گاه L^M در شرایط لم اسپرنر [۲۳] صدق می کند و لذا $|L^M| \leq \binom{n}{\lfloor \frac{n}{2} \rfloor}$ است. حال با توجه به ساختار تیلور و زویکر می توانیم نتیجه گیری کنیم که:

$$d_n \leq |L^M| \leq \binom{n}{\lfloor \frac{n}{2} \rfloor}. \quad (3.2)$$

یکی از هدف های اصلی این پایان نامه بهبودبخشیدن کران بالاست. تعاریف خود را با مثال توضیح خواهیم داد.

مثال ۱.۳.۲. فرض کنید بازی ساده $\Gamma(N, L^M)$ به صورت زیر تعریف شود:

$$N = \{1, 2, 3, 4, 5, 6, 7\}$$

$$L^M = \{\{1, 2, 3\}, \{3, 4, 5, 6\}\}.$$

ائتلاف $\{1, 2\}$ یک ائتلاف بازنده است؛ چون زیرمجموعه ائتلاف ماکسیمال باخت $\{1, 2, 3\}$ است. ائتلاف $\{1, 4\}$ یک ائتلاف برنده می باشد؛ زیرا $\{1, 4\} \not\subseteq \{1, 2, 3\}$ و $\{1, 4\} \not\subseteq \{3, 4, 5, 6\}$. اگر از ساختار تیلور و زویکر استفاده شود، می توان بازی Γ را به عنوان تقاطع دو بازی وزنی زیر نمایش داد:

$$T_1 = \{1, 2, 3\}$$

$$T_2 = \{3, 4, 5, 6\}$$

$$\Gamma = \Gamma_{T_1} \cap \Gamma_{T_2} = (N, W_1 \cap W_2) \implies \Gamma = [1; \circ, \circ, \circ, 1, 1, 1, 1] \cap [1; 1, 1, \circ, \circ, \circ, \circ, 1]$$

بنابراین $d(\Gamma) \leq 2$ است. اگر ثابت کنید که Γ خود به شکل یک بازی وزنی قابل نمایش نیست؛ آن گاه $d(\Gamma) = 2$ نتیجه خواهد شد. این حکم به برهان خلف اثبات می شود.

برهان. فرض کنید Γ یک بازی وزنی با سهم q باشد. ائتلاف های $\{1, 2\}$ و $\{4, 5\}$ چون زیرمجموعه ائتلاف های ماکسیمال باخت L^M هستند، هر دو بازنده می باشند. بنا به تعریف بازی وزنی، بایستی مجموع وزن تمام بازیکنان در یک ائتلاف بازنده کمتر از q باشد و لذا،

$$\begin{cases} w_1 + w_2 < q \\ w_4 + w_5 < q \end{cases} \implies w_1 + w_2 + w_4 + w_5 < 2q.$$

از طرفی، این دو ائتلاف می توانند به گونه ای بازیکن رد و بدل کنند که هر دو ائتلاف ساخته شده برنده شوند. توجه کنید که هر دو ائتلاف $\{1, 4\}$ و $\{2, 5\}$ برنده هستند و لذا مجموع وزن

بازیکنان در این دو ائتلاف بنا به تعریف بازی وزنی باید حداقل $2q$ باشد؛ در این صورت داریم:

$$\begin{cases} w_1 + w_4 \geq q \\ w_2 + w_5 \geq q \end{cases} \implies w_1 + w_2 + w_4 + w_5 \geq 2q.$$

□

که این تناقض است.

۲.۳.۲ کدهای پوششی

یک کد دودویی در حقیقت مجموعه‌ای از $\{0, 1\}$ -بردارها است. یک $\{0, 1\}$ -بردار $x = x_1 x_2 \dots x_n \in \{0, 1\}^n$ می‌تواند به شکل یک ائتلاف $S_x = \{i \in N : x_i = 1\}$ دیده شود. از این لحظه به بعد می‌توان x را به شکل یک بردار و همچنین به عنوان یک ائتلاف در نظر گرفت. از این نقطه نظر استفاده شده و یک کد دودویی به عنوان یک مجموعه از ائتلاف‌ها به منظور دستیابی به ارتباطی بین مفهوم کدهای دودویی و بازی‌های ساده در نظر گرفته می‌شود. فاصله همینگ بین دو $\{0, 1\}$ -بردار، تعداد مؤلفه‌هایی است که دو بردار در آن‌ها متفاوت هستند. به عبارت دیگر؛ فاصله همینگ بین دو ائتلاف x و y به صورت زیر تعریف می‌شود:

$$d(x, y) = |x \setminus y| + |y \setminus x|.$$

تعریف ۶.۳.۲. یک کد پوششی دودویی به طول n و شعاع پوششی ۱ می‌تواند به عنوان یک مجموعه $C \subseteq 2^N$ از ائتلاف‌ها در نظر گرفته شود که هر ائتلافی در فاصله همینگ ۰ یا ۱ از حداقل یکی از عضوهای مجموعه C قرار دارد. به عبارت دیگر؛

$$\forall x \in 2^N \exists c \in C : d(x, c) \leq 1.$$

در این پایان‌نامه K_n نشان‌دهنده کوچک‌ترین اندازه از یک کد پوششی دودویی به طول n با شعاع پوششی ۱ است.

مثال ۲.۳.۲. مجموعه زیر نشان‌دهنده یک کد پوششی دودویی با طول ۴ و شعاع پوششی ۱ است:

$$C = \{\{\}, \{4\}, \{1, 2, 3\}, \{1, 2, 3, 4\}\}.$$

برای مثال، دو ائتلاف $x = \{2, 4\} \notin C$ و $y = \{4\} \in C$ را در نظر بگیرید. توجه کنید که،

$$d(x, y) = |x \setminus y| + |y \setminus x| = 1 + 0 = 1.$$

چون فاصله همینگ بین دو ائتلاف x و y مساوی ۱ است؛ لذا بنا به تعریف کد پوششی دودویی، ائتلاف x توسط ائتلاف y در C پوشش داده می‌شود.

از طرفی، تمام زیرمجموعه‌های $\{1, 2, 3, 4\}$ را نمی‌توان با کمتر از ۴ ائتلاف پوشش داد؛ زیرا هر ائتلاف می‌تواند حداکثر ۵ ائتلاف را در شعاع ۱ خود پوشش دهد؛ ولی $\{1, 2, 3, 4\}$ شامل ۱۶ زیرمجموعه است. بنابراین $K_4 = 4$ می‌باشد.

به‌طور کلی، یک ائتلاف حداکثر $n + 1$ ائتلاف دیگر را که شامل خودش هم می‌شود در شعاع ۱ پوشش می‌دهد؛ بنابراین به حداقل $\frac{2^n}{(n+1)}$ ائتلاف برای یک کد پوششی دودویی با شعاع پوششی ۱ نیاز داریم. لذا $K_n \geq \frac{2^n}{n+1}$ است. کدهای همینگ، کدهای کامل هستند [۲] و در آن‌ها $n = 2^m - 1$ است. همچنین این کدها در کران پایین ذکر شده صدق می‌کنند؛ لذا برای $n = 2^m - 1$ ، $K_n = \frac{2^n}{n+1}$ می‌باشد. برای $n = 2^m$ ، کران پایین $\frac{2^n}{n+1}$ دقیق نیست. در [۲۵] ثابت شده که برای $n = 2^m$ ، $K_n = \frac{2^n}{n}$ است. در کل، تعیین عدد دقیق K_n سخت است؛ ولی می‌توان کران بالای $(n+1)2^n / (ln(n+1) + 1) \leq K_n$ را با استفاده از نتیجه کلاسیک آلون و اسپنسر [۱] برای مجموعه‌های احاطه‌گری در گراف‌ها اثبات کرد.

نتیجه ۱.۳.۲.

$$K_n = \begin{cases} \frac{2^n}{n+1} & ; n = 2^m - 1 \\ \frac{2^n}{n} & ; n = 2^m \\ \leq (ln(n+1) + 1) \frac{2^n}{n+1} & ; o.w. \end{cases}$$

در ادامه، اثبات این نتیجه مورد بررسی قرار می‌گیرد.

قضیه ۱.۳.۲. فرض کنید $G = (V, E)$ گرافی با n رأس و مینیمم درجه $\delta > 1$ باشد؛ آن‌گاه G یک مجموعه احاطه‌گری با حداکثر $n \frac{1+ln(\delta+1)}{\delta+1}$ رأس دارد [۱].

توجه ۱.۳.۲. $\gamma(G_n) = K_n$.

برهان نتیجه ۱.۳.۲. بنا به قضیه ۱.۳.۲ می‌دانیم:

$$\gamma(G) \leq |V(G)| \frac{1 + \ln(\delta + 1)}{\delta + 1}.$$

از آنجا که $\gamma(G_n) = K_n$ است؛ بنابراین داریم:

$$K_n = \gamma(G) \leq 2^n \frac{1 + \ln(n + 1)}{n + 1}.$$

□

۴.۲ اولین کران بالا

از این به‌بعد، هر بازی ساده به‌صورت ائتلاف‌های ماکسیمال باخت تعریف می‌شود. با در نظر گرفتن بازی ساده $\Gamma(N, L^M)$ ، می‌توان یک الگوریتم ارائه داد که Γ به‌صورت حداکثر K_n بازی

وزنی قابل نمایش باشد. در این بخش، گام به گام نشان می‌دهد که چگونه این الگوریتم کار می‌کند. در ادامه توضیحات هر بخش، با یک مثال نشان می‌دهد که هر گام چگونه کار خواهد کرد.

ایده اصلی برای ارائه این الگوریتم، لم زیر است.

لم ۱.۴.۲. اگر $L^M = \bigcup_{i=1}^p L_i$ باشد؛ آن گاه:

$$\Gamma(N, L^M) = \bigcap_{i=1}^p \Gamma(N, L_i).$$

برهان. باید ثابت شود که $\Gamma(N, L^M) \subseteq \bigcap_{i=1}^p \Gamma(N, L_i)$ و $\bigcap_{i=1}^p \Gamma(N, L_i) \subseteq \Gamma(N, L^M)$ می‌باشد. ابتدا $\Gamma(N, L^M) \subseteq \bigcap_{i=1}^p \Gamma(N, L_i)$ ثابت می‌شود. فرض کنید ائتلاف $x \subseteq N$ در $\Gamma(N, L^M)$ بازنده باشد؛ یعنی $y \in L^M$ وجود دارد به طوری که $x \subseteq y$ است. از طرفی $L^M = \bigcup_{i=1}^p L_i$ و لذا حداقل یک i وجود دارد که $y \in L_i$. بنابراین x در $\Gamma(N, L_i)$ بازنده است. در نتیجه ائتلاف x در $\bigcap_{i=1}^p \Gamma(N, L_i)$ نیز بازنده می‌باشد.

حال اثبات می‌شود که $\bigcap_{i=1}^p \Gamma(N, L_i) \subseteq \Gamma(N, L^M)$ است. فرض کنید ائتلاف x در $\bigcap_{i=1}^p \Gamma(N, L_i)$ بازنده باشد؛ لذا $i_0 \in [p]$ چنان وجود دارد به طوری که x در $\Gamma(N, L_{i_0})$ بازنده است؛ یعنی $y \in L_{i_0} \subseteq L^M$ چنان وجود دارد که $x \subseteq y$ می‌باشد. از طرفی داریم: $L_i \subseteq \bigcup_{i=1}^p L_i = L^M$. لذا $y \in L_{i_0} \subseteq L^M$ ، در نتیجه $y \in L^M$. ثابت کردیم y در L^M است و x را شامل می‌شود؛ بنابراین x در $\Gamma(N, L^M)$ بازنده می‌باشد. $\square$

هدف الگوریتم، استفاده از لم ۱.۴.۲ و تقسیم L^M به مجموعه‌های کوچک‌تر است به گونه‌ای که بازی‌های ساخته شده مرتبط، وزنی باشند.

برای مثال، بازی ساده زیر با بازیکنان $\{1, 2, 3, 4\}$ مورد بررسی قرار می‌گیرد.

در این بازی یک ائتلاف برنده است اگر و تنها اگر هر دو بازیکن ۱ و ۲ و یا هر دو بازیکن ۳ و ۴ عضوی از ائتلاف باشند. به عنوان مثال، ائتلاف $\{1, 2, 4\}$ برنده است. از طرفی دیگر، ائتلاف $\{1, 3\}$ بازنده است و در حقیقت یک ائتلاف ماکسیمال باخت می‌باشد؛ چون اگر هر بازیکن دیگر به این ائتلاف بپیوندد تبدیل به یک ائتلاف برنده خواهد شد. به عنوان تذکر، این بازی متعلق به یک کلاس از بازی‌های ساده است که توسط فریکس و پوینت [۱۳] مطالعه شده است.

حال می‌توان توضیح داد که الگوریتم چگونه کار می‌کند:

۱.۴.۲ ورودی

ورودی این الگوریتم یک بازی ساده $\Gamma(N, L^M)$ است.

مثال ۱.۴.۲. بازی ساده $\Gamma_\circ = \Gamma_\circ(N, L^M)$ را به شکل زیر در نظر بگیرید.

$$N = \{1, 2, 3, 4\}$$

$$L^M = \{\{1, 3\}, \{1, 4\}, \{2, 3\}, \{2, 4\}\}.$$

۲.۴.۲ گام ۱

خانوادهٔ ائتلاف‌های $2^N \supset C$ به این صورت ساخته می‌شوند که هر ائتلافی در L^M در نظر گرفته شود، در فاصله همینگ $\circ$ یا ۱ از حداقل یک ائتلاف در مجموعهٔ C قرار داشته باشد. به عبارت دیگر،

$$\forall x \in L^M \quad \exists c \in C : d(x, c) \leq 1.$$

برای نمونه، خانوادهٔ C در مثال ۱.۴.۲ به شکل زیر است.

$$C = \{\{4\}, \{1, 2, 3\}\}.$$

ملاحظه ۱.۴.۲. توجه کنید که این خانواده لزوماً منحصر بفرد نیست.

۳.۴.۲ گام ۲

در این مرحله قصد داریم با استفاده از خانوادهٔ C که در گام اول ساخته شد، مجموعهٔ L^M را افزایش کنیم.

فرض کنید $\{L_c\}_{c \in C}$ یک افراز از L^M باشد؛ به طوری که تمام عضوهای L_c ، فاصله همینگ $\circ$ یا ۱ با ائتلاف‌های $c \in C$ داشته باشند. به عبارتی داریم:

$$\forall x \in L_c : d(x, c) \leq 1.$$

توجه کنید که بنا به لم ۱.۴.۲، $\Gamma(N, L^M) = \bigcap_{c \in C} \Gamma(N, L_c)$ است.

با استفاده از کد پوششی که در گام دوم به دست آمد، مجموعهٔ L^M در مثال ۱.۴.۲ با ساختن مجموعه‌های L_c ، به دو مجموعهٔ زیر افراز می‌شود.

$$L_{\{4\}} = \{\{1, 4\}, \{2, 4\}\}$$

$$L_{\{1, 2, 3\}} = \{\{1, 3\}, \{2, 3\}\}.$$

۴.۴.۲ گام ۳

برای هر $c \in C$ ، بازی $\Gamma(N, L_c)$ به عنوان یک بازی وزنی $[q^c; w_1^c, w_2^c, \dots, w_n^c]$ تعریف می‌شود. در لم (۱.۵.۲) ثابت می‌شود که $\Gamma(N, L_c)$ برای هر $c \in C$ وزنی است و جزئیات چگونگی محاسبه وزن‌ها و بخش‌ها را نیز توضیح خواهیم داد. در ادامه، $\Gamma(N, L_c)$ برای هر یک از مجموعه‌های L_c در مثال ۱.۴.۲ به صورت زیر می‌باشد.

$$\Gamma_\circ(N, L_{\{1\}}) = [2; 1, 1, 2, \circ]$$

$$\Gamma_\circ(N, L_{\{1, 2, 3\}}) = [2; 1, 1, \circ, 2].$$

۵.۴.۲ خروجی

در نهایت، از لم (۱.۴.۲) استفاده می‌شود و Γ به عنوان تقاطع بازی‌های وزنی که در گام سوم ساخته شد، تعریف می‌شود:

$$\Gamma = \bigcap_{c \in C} [q^c; w_1^c, w_2^c, \dots, w_n^c].$$

این به این معنی است که توانسته‌ایم Γ را به صورت اشتراک $|C|$ بازی وزنی معرفی کنیم. پیرو مثال ۱.۴.۲ داریم:

$$\Gamma_\circ = [2; 1, 1, 2, \circ] \cap [2; 1, 1, \circ, 2].$$

توجه کنید که $|C|$ هرچقدر کوچک‌تر باشد؛ یعنی توانسته‌ایم Γ را به صورت اشتراک تعداد کمتری بازی وزنی نمایش دهیم. از طرفی، هر کد پوششی با شعاع ۱ می‌تواند به عنوان C مورد استفاده قرار گیرد. لذا Γ را می‌توان به صورت اشتراک حداکثر K_n بازی وزنی نمایش داد. به عبارت دیگر، $d_n \leq K_n$ می‌باشد. بنابراین با به کار بردن کران‌های مربوط به K_n که قبلاً بیان شد، کران‌های زیر را برای d_n داریم.

$$d_n \leq K_n = \frac{2^n}{n+1} \quad ; \quad n = 2^m - 1 \quad (۴.۲)$$

$$d_n \leq K_n = \frac{2^n}{n} \quad ; \quad n = 2^m \quad (۵.۲)$$

$$d_n \leq K_n \leq (\ln(n+1) + 1) \frac{2^n}{n+1} \quad ; \quad n \in \mathbb{N}. \quad (۶.۲)$$

جدول ۱.۲: این جدول کران‌های بالا و پایین برای d_n توأم با نتایج [۲۴] و [۲۵] را نشان می‌دهد.

n	کران پایین	کران بالا	$1 - \binom{n}{\lfloor n/2 \rfloor}$
۶	۴	۱۲	۱۹
۷	۷	۱۶	۳۴
۸	۱۴	۳۲	۶۹
۹	۱۸	۶۲	۱۲۵
۱۰	۳۶	۱۲۰	۲۵۱
۱۱	۶۶	۱۹۲	۴۶۱
۱۲	۱۳۲	۳۸۰	۹۲۳
۱۳	۱۶۶	۷۰۴	۱۷۱۵
۱۴	۳۲۵	۱۴۰۸	۳۴۳۱
۱۵	۵۸۵	۲۰۴۸	۶۴۳۴

توجه کنید که در هر ۳ حالت فوق، کران‌های بالا به‌صورت قابل ملاحظه‌ای کمتر از $\binom{n}{\lfloor n/2 \rfloor}$ است. با توجه به روابط ۲.۲ و ۳.۲، کران‌های بالای سطر اول و دوم جدول، نمایانگر پیشرفت بر روی فاکتور $\sqrt{n}$ می‌باشند و همه کران‌ها $o(\binom{n}{\lfloor n/2 \rfloor})$ است.

این موضوع اهمیت دارد که استفاده از یک کد پوششی دودویی به‌عنوان یک راه‌حل کلی ممکن است به نتیجه مطلوبی ختم نشود؛ زیرا در این روش، ساختار L^M برای بازی خاص مورد توجه قرار نگرفته است.

استرگارد و کایکون [۲۵] بعضی از کران‌های بالای K_n را لیست کرده‌اند و همچنین آن‌ها را می‌توانیم به‌عنوان کران‌های بالا برای d_n نیز استفاده کنیم. جدول ۱.۲ کران‌های بالا و پایین [۲۴] را نشان می‌دهد.

۵.۲ اثبات اولین کران بالا

در این بخش، الگوریتم ارائه‌شده مورد بررسی و تحلیل قرار می‌گیرد و کران بالای ادعا شده به‌عنوان یک قضیه ثابت می‌شود.

در این راستا باید ثابت شود که الگوریتم ارائه‌شده قادر است هرگونه بازی ورودی را به‌عنوان تقاطع بازی‌های وزنی بیان کند. واضح است که این الگوریتم، مجموعه C در گام اول را ایجاد می‌کند و تقسیم L^M را نیز در گام دوم می‌تواند بسازد. از آنجا که این الگوریتم از روش تجزیه پیشنهادی در لم (۱.۴.۲) استفاده می‌کند؛ بنابراین کافی است ثابت شود که تمام بازی‌های فرض شده در گام سوم وزنی می‌باشند.

فرض کنید $\Gamma(N, L^M)$ یک بازی ساده است و C همان مجموعه معرفی‌شده در گام اول الگوریتم باشد. در این صورت،

لم ۱.۵.۲. $\Gamma(N, L_c)$ برای هر $c \in C$ وزنی است.

برهان. فرض کنید $c \in C$ باشد؛ در این صورت داریم:

$$L_c = \{x \in L^M \mid d(x, c) \leq 1\}.$$

از آنجا که تمام اعضای L_c ائتلاف‌های ماکسیمال باخت می‌باشند؛ لذا بنا به ملاحظه ۱.۳.۲ هیچ دو عضوی از L_c وجود ندارد که یکی از آن‌ها شامل دیگری باشد. توجه کنید که هر عضو در L_c ، زیرمجموعه c و یا ابرمجموعه c می‌باشد. از طرفی، همه اعضای L_c نسبت به زیرمجموعه‌بودن، ماکسیمال هستند؛ زیرا $L_c \subset L^M$ است. بنابراین در L_c یکی از ۳ حالت زیر امکان‌پذیر است:

$$1. \forall x \in L_c : x \subset c$$

$$2. L_c = \{c\}$$

$$3. \forall x \in L_c : c \subset x$$

در این قسمت نشان خواهیم داد که چگونه $\Gamma(N, L_c)$ به عنوان یک بازی وزنی در هر ۳ حالت قابل بیان است.

حالت ۱: مجموعه L_c شامل ائتلاف‌هایی است که دقیقاً از حذف یک عامل از c برای هر عضو L_c به وجود می‌آیند. فرض کنید R مجموعه عامل‌های حذف‌شده باشد: $R = \cup_{x \in L_c} (c \setminus x)$. همچنین فرض کنید S یک مجموعه برنده، مشمول در c باشد. برای هر $x \in L_c$ ، می‌دانیم که S زیرمجموعه x نیست؛ زیرا x یک ائتلاف ماکسیمال باخت است و بنا به تعریف ائتلاف ماکسیمال باخت، همه زیرمجموعه‌های ائتلاف ماکسیمال باخت، بازنده می‌باشد. بنابراین S باید شامل آن عاملی باشد که از c حذف شده است، تا x را شکل دهد. لذا S باید شامل تمام اعضای حذف‌شده، یعنی شامل R باشد تا برنده شود، زیرا اگر حتی یک عضو از اعضای حذف‌شده را نداشته باشد؛ آن‌گاه زیرمجموعه یکی از اعضای L_c می‌باشد. بنابراین اگر $R \subseteq S$ باشد؛ آن‌گاه S زیرمجموعه هیچ‌یک از اعضای L_c ، که ائتلاف‌های ماکسیمال باخت هستند، نیست و بنابراین S برنده می‌باشد.

حال اگر $c \not\subseteq S$ ، از آنجا که S ائتلاف برنده است؛ لذا برای هر $x \in L_c$ ، $S \not\subseteq x$. بنابراین $S \setminus c \neq \emptyset$. به عبارت دیگر، S در $\Gamma(N, L_c)$ برنده است اگر و تنها اگر $S \setminus c \neq \emptyset$ یا $R \subseteq S$ باشد. حال ثابت می‌کنیم که می‌توان $\Gamma(N, L_c)$ را به عنوان یک بازی وزنی با $q = |R|$ در نظر گرفت. برای این هدف ابتدا باید وزن بازیکنان مشخص شود.

بازیکنی که در همه ائتلاف‌های مجموعه L_c حضور دارد مانند عضو خنثی عمل می‌کند؛ یعنی حضور یا عدم حضور این بازیکن در برد و باخت بازی تأثیری ندارد، در نتیجه وزن صفر به این بازیکن تعلق می‌گیرد. وزن هر بازیکنی که خارج از c قرار می‌گیرد؛ یعنی $i \notin c$ ، را $|R|$ در نظر می‌گیریم؛ زیرا ائتلافی برنده است که مشمول در L_c نباشد. برای این امر کافی است، تنها یک عضو خارج از c داشته باشد. می‌دانیم ائتلافی که شامل تمام اعضای R باشد یک ائتلاف برنده است. لذا بنا به تعریف بازی وزنی، بایستی مجموع وزن بازیکنان این ائتلاف، بیشتر یا مساوی $|R|$ باشد. در نتیجه اگر به هر بازیکنی که در R قرار می‌گیرد وزن یک بدهیم، ائتلاف برنده می‌باشد. بنابراین وزن‌دهی بازیکنان به صورت زیر می‌باشد.

• اگر $i \notin c$ ، آن‌گاه $w_i = |R|$ می‌باشد.

• اگر $i \in R$ باشد، آن‌گاه $w_i = 1$ می‌باشد.

• برای هر بازیکن باقی‌مانده $w_i = 0$ می‌باشد.

اکنون به سادگی می‌توان بررسی کرد که این وزن‌دهی همان وزن‌دهی مطلوب مدنظر است. برای روشن‌تر شدن نحوه عمل، مثال زیر ارائه می‌شود.

مثال ۱.۵.۲. بازی $\Gamma(N, L_c)$ با مجموعه‌های زیر را در نظر بگیرید:

$$N = \{1, 2, 3, 4, 5\}$$

$$c = \{1, 2, 3, 4\}$$

$$L_c = \{\{1, 2, 3\}, \{1, 2, 4\}, \{1, 3, 4\}\}.$$

برای این بازی $R = \{2, 3, 4\}$ و $\Gamma(N, L_c) = [3; 0, 1, 1, 3]$ می‌باشد.

حالت ۲: بررسی این حالت بسیار ساده است. در این حالت از بازی وزنی با بخش $q = 1$ استفاده می‌شود که وزن ۰ به تمام بازیکنان c و وزن ۱ به سایر بازیکنان تعلق می‌گیرد.

حالت ۳: تمام ائتلاف‌های L_c با اضافه کردن دقیقاً یک عامل به c ساخته می‌شوند. فرض کنید A مجموعه عامل‌های افزوده باشد: $A = \bigcup_{x \in L_c} (x \setminus c)$. اگر ائتلاف S برنده باشد و شامل هیچ‌یک از بازیکنان $(c \cup A)^c$ نباشد، باید حداقل شامل دو بازیکن از A باشد (در غیر این صورت S بازنده خواهد بود). متقابلاً، S برنده است اگر شامل یک بازیکن خارج از $c \cup A$ و یا حداقل دو بازیکن از A باشد. این نشان می‌دهد که $\Gamma(N, L_c)$ می‌تواند به صورت بازی‌های وزنی با بخش $q = 2$ و توزیع وزنی زیر بیان شود:

اگر ائتلافی، شامل حتی یک عضو خارج از $c \cup A$ باشد، برنده است. لذا به هر بازیکنی که خارج از $c \cup A$ قرار می‌گیرد وزن ۲ می‌دهیم. به بازیکنی که در A حضور دارد، وزن ۱ اختصاص داده می‌شود. زیرا همان‌طور که می‌دانیم، یک ائتلاف برنده است اگر و تنها اگر حداقل شامل دو

بازیکن از A باشد. در نهایت وزن صفر به بازیکنی تعلق می‌گیرد که در همه ائتلاف‌های L_c حضور دارد. به عبارتی داریم:

• اگر $i \notin c \cup A$ ، آن‌گاه $w_i = 2$ می‌باشد.

• اگر $i \in A$ ، آن‌گاه $w_i = 1$ می‌باشد.

• اگر $i \in \bigcap_{x \in L_c} x$ ، آن‌گاه $w_i = 0$ می‌باشد.

مثال ۲.۵.۲. فرض کنید $\Gamma(N, L_c)$ با مجموعه‌های زیر باشد:

$$N = \{1, 2, 3, 4, 5, 6, 7\}$$

$$c = \{1, 2, 3\}$$

$$L_c = \{\{1, 2, 3, 4\}, \{1, 2, 3, 5\}, \{1, 2, 3, 6\}\}.$$

بنابراین $A = \{4, 5, 6\}$ و $\Gamma(N, L_c) = [2; 0, 0, 0, 1, 1, 1, 2]$ می‌باشد.

□

حال به بیان اصلی‌ترین نتیجه این فصل پرداخته می‌شود.

قضیه ۱.۵.۲. فرض کنید $\Gamma(N, L^M)$ یک بازی ساده و $C \subset 2^N$ یک مجموعه از ائتلاف‌ها باشد به‌طوری‌که:

$$\forall x \in L^M \quad \exists c \in C : d(x, c) \leq 1.$$

در این صورت بُعد $\Gamma(N, L^M)$ از بالا به $|C|$ محدود می‌شود.

برهان. بدیهی است که الگوریتم ارائه‌شده می‌تواند برای معرفی Γ به عنوان تقاطعی از $|C|$ بازی وزنی مورد استفاده قرار گیرد. لم ۱.۴.۲ و لم ۱.۵.۲ درست بودن الگوریتم را تضمین می‌کنند. در ابتدا $C \subset 2^N$ به گونه‌ای انتخاب می‌شود که برای هر $x \in L^M$ ، $c \in C$ چنان وجود دارد که $d(x, c) \leq 1$ باشد. بنابر لم ۱.۵.۲، برای هر $c \in C$ ، $\Gamma(N, L_c)$ یک بازی وزنی است. حال ثابت می‌کنیم $L^M = \bigcup_{c \in C} L_c$ است. برای این منظور باید ثابت کنیم $L^M \subseteq \bigcup_{c \in C} L_c$ و $\bigcup_{c \in C} L_c \subseteq L^M$.

با توجه به تعریف L_c ها، واضح است که $\bigcup_{c \in C} L_c \subseteq L^M$ می‌باشد. اکنون ثابت می‌کنیم $L^M \subseteq \bigcup_{c \in C} L_c$ است. فرض کنید $x \in L^M$ دلخواه باشد. بنا به تعریف، $c \in C$ چنان وجود دارد که $d(x, c) \leq 1$ است و لذا $x \in L_c$ می‌باشد. بنابراین $L^M \subseteq \bigcup_{c \in C} L_c$. در نتیجه با استفاده از لم ۱.۴.۲، داریم $\Gamma(N, L^M) = \bigcap_{c \in C} \Gamma(N, L_c)$. از آنجا که هر $\Gamma(N, L_c)$ یک بازی وزنی است؛ لذا $d_n \leq |C|$

□

توجه کنید حالت خاص $C = L^M$ به کران بالای $|L^M|$ معرفی شده توسط تیلور و زویکر [۲۹] منجر می شود.

اگر یک کد پوششی دودویی با طول n و شعاع پوششی ۱ را جایگزین مجموعه C در قضیه ۱.۵.۲ کنیم، نتیجه زیر حاصل می شود:

نتیجه ۱.۵.۲.

$$d_n \leq K_n.$$

برهان. فرض کنید $\Gamma(N, L^M)$ یک بازی وزنی با $N = \{1, 2, \dots, n\}$ باشد. $C \subset 2^N$ به گونه ای انتخاب می شود که برای هر $x \in L^M$ ، $z \in C$ چنان وجود دارد که $d(x, z) \leq 1$ ، لذا $d_n \leq |C|$ می باشد. اگر $C \subset 2^N$ یک کد پوششی مینیمم باشد؛ آن گاه برای هر بازی $\Gamma(N, L^M)$ داریم:

$$\forall x \in L^M \exists c \in C \text{ s.t. } d(x, c) \leq 1.$$

$$\text{لذا } d_n \leq |C| = K_n.$$

□

باید در نظر داشت که تنها به خانواده C برای پوشش مجموعه L^M در قضیه ۱.۵.۲ نیاز است. ما قادر خواهیم بود که ساختار L^M را به منظور دستیابی به کران بالای بهتر نسبت به نتیجه فوق که در آن مجموعه اصلی، تمام ائتلاف های ممکن را پوشش می دهد، به کار ببریم. به عنوان مثال، می توان از این حقیقت که L^M از خانواده اسپرر است استفاده کرد که هیچ عضوی شامل عضو دیگری از خانواده نیست. این موضوع توضیح می دهد که کران $d_n \leq K_n$ به عنوان یک نتیجه انتخاب شد تا نتیجه گیری قوی تری حاصل شود.

۶.۲ کران بالای دوم

در این قسمت یک بار دیگر ایده به کاررفته در اثبات لم ۱.۴.۲ مورد استفاده قرار می گیرد و یک کران بالای دیگر که تعمیم دهنده کران بالای مبتنی بر $|L^M|$ اثبات شده توسط تیلور و زویکر [۲۹] است، ارائه می شود. این کران بالا در ارتباط با کد SECDED است که یک کد دودویی با کمترین فاصله حداقل ۴ می باشد.

قضیه ۱.۶.۲. فرض کنید $\Gamma(N, L^M)$ یک بازی ساده باشد. خانواده $C \subseteq L^M$ از ائتلاف های ماکسیمال بخت وجود دارد؛ به گونه ای که برای هر $x \neq y \in C$ داریم $d(x, y) \geq 4$ و بُعد $\Gamma(N, L^M)$ حداکثر $\frac{1}{4}(|L^M| + |C|)$ است.

برهان. مجموعه ماکسیمال $M \subseteq L^M \times L^M \setminus \{(x, x); x \in L^M\}$ را چنان در نظر بگیرید که اولاً، برای هر $(x, y) \in L^M$ ، $d(x, y) \leq 3$ باشد و علاوه بر این، هر $x \in L^M$ در حداکثر یک زوج مرتب

از M (چه به عنوان درایه اول و چه به عنوان درایه دوم) ظاهر شده باشد. باید نشان داد که برای هر $(x, y) \in L^M$ بازی ساده $\Gamma(N, \{x, y\})$ یک بازی وزنی است. سه حالت $d(x, y) = 1$ ، $d(x, y) = 2$ و $d(x, y) = 3$ را در نظر بگیرید.

حالت اول: $d(x, y) = 1$

توجه کنید که در این حالت، $x \subset y$ یا $y \subset x$ است؛ ولی بنا به اسپرنر بودن L^M چنین چیزی امکان پذیر نمی باشد. (دقت کنید که $(x, y) \in L^M$).

حالت دوم: $d(x, y) = 2$

با توجه به اسپرنر بودن L^M باید داشته باشیم: $x \setminus y = \{a\}$ و $y \setminus x = \{b\}$.

می خواهیم نشان دهیم که این بازی ساده را می توان به صورت یک بازی وزنی نمایش داد. توجه کنید که در بازی $\Gamma(N, \{x, y\})$ ، هر ائتلافی که زیرمجموعه x و یا هر ائتلافی که زیرمجموعه y باشد، بازنده است. حال بازی $\Gamma(N, \{x, y\})$ به عنوان یک بازی وزنی با سهم $q = 2$ به شکل زیر معرفی می شود.

بازیکنان در $x \cap y$ وزن ۰ می گیرند؛ زیرا بازیکنی که در همه ائتلاف های ماکسیمال باخت حضور دارد مانند عضو خنثی عمل می کند؛ لذا حضور یا عدم حضور این بازیکن در برد و باخت بازی تأثیری ندارد. به طور کلی، وزن دهی بازیکنان به شکل زیر تعریف می شود:

$$w_i = \begin{cases} 0 & ; i \in x \cap y \\ 1 & ; i \in \{a, b\} \\ 2 & ; o.w. \end{cases}$$

اگر یک ائتلاف دلخواه بازنده در بازی $\Gamma(N, \{x, y\})$ در نظر بگیرید، باید زیرمجموعه ائتلاف x یا ائتلاف y باشد؛ در این حالت، ائتلاف حداکثر وزن ۱ می گیرد؛ چون حداکثر یک عضو از بین a و b را دارد. از طرفی، ائتلافی برنده است که زیرمجموعه هیچ کدام از ائتلاف های x و y نباشد. اگر ائتلافی زیرمجموعه x و y نباشد، باید بازیکنی خارج از ائتلاف های x و y داشته باشد که در این حالت حداقل وزن ۲ می گیرد و لذا در بازی وزنی نیز برنده است، اگر عضوی خارج از x و y نداشته باشد؛ در این صورت باید هر دو عضو a و b را دارا باشد که در این حالت نیز وزن ۲ را می گیرد و باز هم در بازی وزنی برنده است. در حالت کلی، اگر ائتلافی یک عضو خارج از $x \cup y$ یا هر دو عضو a و b را داشته باشد، حداقل وزنی که به این ائتلاف اختصاص داده می شود ۲ می باشد.

حالت سوم: $d(x, y) = 3$

با توجه به اسپرنر بودن L^M و بدون از دست دادن کلیت، می توان فرض کرد که $x \setminus y$ شامل یک عضو و $y \setminus x$ شامل دو عضو باشد ($y \setminus x = \{a, b\}$ و $x \setminus y = \{c\}$).

بازی ساده $\Gamma(N, \{x, y\})$ به صورت یک بازی وزنی با سهم $q = 3$ و وزن های زیر معرفی می شود:

$$w_i = \begin{cases} 0 & ; i \in x \cap y \\ 1 & ; i \in \{a, b\} \\ 2 & ; i = c \\ 3 & ; o.w. \end{cases}$$

به سادگی می توان دید که یک ائتلاف برنده است اگر و تنها اگر یکی از حالت های زیر رخ دهد:
(۱) ائتلاف شامل حداقل یک بازیکن در $N \setminus (x \cup y)$ باشد.

(۲) ائتلاف شامل یک بازیکن واقع در $x \setminus y$ و حداقل یکی از دو بازیکن واقع در $y \setminus x$ باشد.
حال اگر یک ائتلاف برنده باشد با توجه به وزن دهی، اگر در حالت (۱) صدق کند حداقل وزن آن ۳ است و اگر در حالت (۲) صدق کند نیز حداقل وزن برابر ۳ خواهد بود. همچنین اگر ائتلاف بازنده باشد؛ آن گاه چون در دو حالت بالا صدق نمی کند، حداکثر مجموع وزن نسبت داده شده به اعضای آن ۲ است.

نتیجه می گیریم بازی $\Gamma(N, \{x, y\})$ یک بازی وزنی می باشد.

برای واضح تر شدن اثبات در این قسمت مثال زیر را مشاهده کنید.

مثال ۱.۶.۲. ساختار را با یک مثال با مجموعه های $N = \{1, 2, 3, 4, 5, 6, 7\}$ و $x = \{1, 2, 3, 4\}$ و $y = \{2, 3, 5\}$ نشان می دهیم. وزن مرتبط با بازی $[3; 1, 0, 0, 1, 2, 3, 3]$ است.

فرض کنید C مجموعه ائتلاف هایی باشد که در M با هیچ ائتلافی جفت نشده اند. از آنجا که M ماکسیمال است، هر دو ائتلاف در C دارای فاصله حداقل ۴ هستند؛ لذا برای هر $x \neq y \in C$ داریم، $d(x, y) \geq 4$.

از طرفی، هر عضو از مجموعه L^M یا در M است یا در C . به عبارت دیگر، داریم:

$$C \cup (\cup_{(x,y) \in M} \{x, y\}) = L^M.$$

حال نشان می دهیم $|M| = \frac{|L^M| - |C|}{4}$ است. توجه کنید که اعضای C اعضای L^M هستند که در هیچ زوج مرتبی از M نیامده اند. اگر این اعضا را از L^M حذف کنیم، اعضای باقی مانده آن هایی هستند که در زوج مرتب ها حضور دارند. چون عضوی در دو زوج مرتب نیامده است، پس اعضای $L^M \setminus C$ را می توان دوبه دو جفت کرد و لذا $|M| = \frac{|L^M| - |C|}{4}$ می باشد. از طرفی، بنا بر لم ۱.۴.۲ داریم،

$$\Gamma(N, L^M) = (\bigcap_{(x,y) \in M} \Gamma(N, \{x, y\})) \cap (\bigcap_{c \in C} \Gamma(N, \{c\})).$$

در نتیجه $d_n \leq |M| + |C| \leq \frac{|L^M|}{4} - \frac{|C|}{4} + |C| = \frac{|L^M| + |C|}{4}$ است.

□

نتیجه ۱.۶.۲.

$$d_n \leq |L^M|.$$

برهان. فرض کنید $C \subseteq 2^n$ باشد، داریم:

$$d_n \leq \min\{|C| \mid \forall x \in L^M \exists c \in C \text{ s.t. } d(x, c) \leq 1\} \leq |L^M|.$$

□

از قضیه ۱.۶.۲ نتیجه زیر حاصل می‌شود.

نتیجه ۲.۶.۲. اگر L^M کد SECDED نباشد، بُعد $\Gamma(N, L^M)$ کمتر از $|L^M|$ است.

برهان. با توجه به اثبات قضیه ۱.۶.۲، اگر L^M یک کد SECDED نباشد؛ آن‌گاه $M \neq \emptyset$ و لذا $C \subsetneq L^M$ می‌باشد؛ بنابراین $|C| \leq |L^M|$ است. داریم:

$$d_n \leq \frac{1}{2}(|L^M| + |C|) < \frac{1}{2}(|L^M| + |L^M|) = |L^M|.$$

□

فصل ۳

بازی کلاه

نظریه بازی‌ها^۱ و کاربردهای آن زیرمجموعه‌ای از علم ریاضیات است که می‌کوشد با استفاده از طراحی و تحلیل سناریو، رفتارها و نتایج تصمیم‌گیری موجوداتی را که حق انتخاب دارند، در تعامل با یکدیگر پیش‌بینی کند.



بعید است بتوانید مثالی کلاسیک‌تر و ساده‌تر از بازی سنگ، کاغذ، قیچی پیدا کنید که مصداقی شناخته‌شده از این نوع فضای تعاملی باشد (چند بازیگر، چند قانون، حق انتخاب و نتایج مشخص برای هر وضعیت).

نظریه بازی‌ها می‌کوشد شرایط پیچیده در تعامل حیوانات، انسان‌ها، سازمان‌ها، کسب و کارها، اقتصادها و کشورها را تا حدی ساده کند. سپس می‌کوشد با تشخیص گزینه‌های موجود، منابعی که کمیاب هستند، اهداف و اولویت‌های کسانی که درگیر بازی هستند و قواعد بازی، دستاوردهای بازی و احتمال وقوع هر کدام را تا حد امکان پیش‌بینی کند. با توجه به تعریفی که ارائه شد، هر جا که منابع محدود، گزینه‌های مختلف تصمیم‌گیری، دستاوردهای متفاوت در اثر انتخاب‌های متفاوت و امکان همکاری یا رقابت بین بازیگران وجود داشته باشد، می‌توان از نظریه بازی‌ها برای درک و تحلیل بهتر شرایط موجود استفاده

¹Game theory

کرد.

نظریه بازی‌ها یکی از علوم تصمیم‌گیری است (مانند تحقیق در عملیات، اقتصاد مهندسی و ...) با این تفاوت که به مطالعه مسائلی می‌پردازد که در آن‌ها نتایج یک تصمیم وابسته به تصمیمات دو یا چند تصمیم‌گیرنده مستقل است (بازیکنان). به بیان دیگر هیچ‌کدام از تصمیم‌گیرندگان کنترل کاملی بر نتایج تصمیمات خود ندارند.

تعیین استراتژی‌های رقابتی، همکاری، و یا ترکیبی از این‌ها در همه سازمان‌ها (تجاری، صنعتی، اجتماعی، سیاسی، نظامی و ...) از مباحث اساسی است و نظریه بازی‌ها به پشتوانه نبوغ بنیان‌گذاران آن، ابزارهای لازم را در اختیار تصمیم‌سازان و پشتیبانان آن‌ها قرار می‌دهد. یازده نفر از کسانی که تاکنون موفق به دریافت جایزه نوبل شده‌اند، در حوزه نظریه بازی‌ها فعالیت داشته‌اند.

۱.۳ معرفی

پازل کلاه در مقاله مارتین گاردنر [۱۴] در سال ۱۹۶۱ معرفی شده است. این تعریف، الهام گرفته شده از رساله دکتری ابرت (۱۹۹۵) است [۸]. در این خصوص بولر [۴] اظهار داشت: «قابل توجه است که یک مسئله صرفاً تفریحی به مرزهای تحقیق بسیار نزدیک است.» همچنین مقالاتی در مجله نیویورک تایمز^۱ [۲۷]، دایزیت^۲ [۳] و ای‌بی‌سی نیوز^۳ [۲۶] درباره این موضوع توجه گسترده‌ای را به خود جلب کرده‌اند. برای جلب توجه بیشتر قسمتی از یک مقاله چاپ شده در مجله نیویورک تایمز را در ادامه خواهیم دید.

مسئله کلاه این‌گونه است:

سه بازیکن وارد اتاق می‌شوند و یک کلاه قرمز یا آبی روی سر هر فرد قرار می‌گیرد. رنگ هر کلاه توسط یک سکه تعیین می‌شود و نتیجه پرتاب سکه بدون تأثیر بر روی دیگران است. هر کسی می‌تواند کلاه بازیکنان دیگر را ببیند؛ اما رنگ کلاه خودش را نمی‌تواند ببیند. هیچ‌گونه ارتباطی بین بازیکنان، به جز جلسه استراتژی قبل از شروع بازی، مجاز نیست. آن‌ها فقط یک مرتبه برای نگاه کردن به کلاه بازیکنان دیگر فرصت دارند، بازیکنان باید در یک زمان (با هم) رنگ کلاه‌های خود را حدس بزنند و یا پاس دهند. گروهی برنده جایزه سه میلیون دلاری است که حداقل یک بازیکن به درستی حدس بزند و هیچ بازیکنی حدس اشتباه نزند. همان بازی را می‌توان با تعداد زیادی از بازیکنان بازی کرد. مشکل عمومی، یافتن یک استراتژی برای گروه است، که شانس خود را برای کسب جایزه به حداکثر برساند. برای مثال، یک استراتژی واضح برای بازیکنان این‌گونه خواهد بود که یک بازیکن همیشه یک رنگ ثابت

¹New York Times

²Die Zeit

³abcNews

را حدس بزند (مثلاً در هر مرحله از بازی فقط رنگ «قرمز» را حدس بزند)، در حالی که بازیکنان دیگر «پاس» دهند. این استراتژی شانس برنده شدن گروه را تا ۵۰ درصد افزایش می‌دهد. آیا گروه می‌تواند بهتر عمل کند؟

از نظر ریاضی‌دانان، جذابیت این مسئله به این دلیل است که این مسئله تنها یک معما برای سرگرمی نیست که آن را حل کرده و دیگر با آن کاری نداشته باشیم. در عوض، این مسئله ارتباطات عمیق و غیرمنتظره‌ای با نظریه کدگذاری، یک حیطه فعال در ریاضی با کاربرد وسیع در ارتباطات مخابراتی و علوم رایانه‌ای دارد. ریاضی‌دانان در تلاش‌های خود برای یافتن یک راه حل کامل برای این مسئله، تئوری‌های جدیدی را در نظریه کدگذاری اثبات می‌کنند که ممکن است کاربردهایی فراتر از پازل‌های ریاضی داشته باشند.

در این فصل، مسئله کلاه ابرت در هر دو حالت متقارن و نامتقارن مورد بررسی قرار می‌گیرد: بر سر N بازیکن متمایز، به‌طور تصادفی کلاه‌های سفید یا سیاه قرار می‌گیرد. احتمال قرارگرفتن کلاه سفید و یا سیاه به ترتیب p و q است ($p + q = 1$). توجه کنید که حالت $p \neq q$ ممکن است؛ ولی همواره p و q مشخص و برای همه بازیکنان ثابت می‌باشد. همه بازیکنان به‌طور هم‌زمان رنگ کلاه خود را حدس می‌زنند. تنها اطلاعاتی که آن‌ها برای این کار در اختیار دارند؛ رنگ کلاه $N - 1$ بازیکن دیگر است. لازم به ذکر است که هر بازیکن می‌تواند به جای حدس زدن رنگ کلاهش، پاس دهد؛ یعنی هیچ رنگی حدس نزند. تیمی برنده است که حداقل یک بازیکن رنگ کلاه خود را به درستی حدس بزند و هیچ بازیکنی حدس اشتباه نزند. هدف ما به حداکثر رساندن احتمال برنده شدن در بازی و توصیف استراتژی‌هایی برای این منظور است. مسئله کلاه متقارن ($p = q = 0.5$) با $N = 2^k - 1$ بازیکن در [۹]، با استفاده از کدهای همینگ، و با $N = 2^k$ بازیکن در [۷]، با استفاده از کدهای همینگ افزوده حل شده است. گیو و همکارانش [۱۵] مسئله کلاه سه نفر را مطرح کردند که در آن امکان رخداد ۸ رنگ - وضعیت دارای احتمالات متفاوت $p_1, \dots, p_8$ هستند. آن‌ها یک برنامه کامپیوتری را برای فهرست کامل از تمام ۵۳۱۴۴۱ استراتژی ممکن با مجموعه‌های شامل موقعیت‌های برنده شدن نوشته‌اند. بررسی این لیست، ۱۲ استراتژی ماکسیمال را ارائه می‌دهد. به سادگی احتمال پیروزی در هریک از این ۱۲ استراتژی را محاسبه و یک استراتژی را با حداکثر احتمال انتخاب می‌کند.

بورک و همکارانش [۵] سعی کردند مسئله کلاه در حالت متقارن را با $N = 3, 4, 5, 7$ بازیکن با استفاده از برنامه‌نویسی ژنتیکی حل کنند. آن‌ها به این نتیجه رسیدند که؛ پازل N - زندانی (نام‌های جایگزین: مسئله کلاه، بازی کلاه) به برنامه‌نویسی ژنتیکی چالش‌های جدیدی برای غلبه کردن، ارائه می‌دهد.

لنسترا و سروسی [۲۶] نشان می‌دهند که در مورد دو رنگ کلاه و برای هر مقدار N ، استراتژی بازی برابر با کدهای پوششی دودویی از شعاع ۱ است.

از ترکیب نتایج لنسترا و سروس با جداول برای کران‌ها روی کدهای پوششی [۱۷]، نتایج زیر به دست می‌آید:

N	۲	۳	۴	۵	۶	۷	۸	۹
$K(N, 1)$	۲	۲	۴	۷	۱۲	۱۶	۳۲	۶۲
ماکزیمم احتمال	$1 - \frac{2}{2^2}$	$1 - \frac{2}{2^3}$	$1 - \frac{2}{2^4}$	$1 - \frac{2}{2^5}$	$1 - \frac{2}{2^6}$	$1 - \frac{2}{2^7}$	$1 - \frac{2}{2^8}$	$1 - \frac{2}{2^9}$

یادآوری: $K(N, 1)$ برابر کوچک‌ترین اندازه یک کد پوششی دودویی از شعاع ۱ می‌باشد.

توجه: تمام نتایج جدول فوق برای بازی متقارن معتبر هستند.

یک کران پایین برای $K(9, 1)$ در سال ۲۰۰۱ توسط استرگاد-پلاس^۱ و کران بالا در سال ۲۰۰۵ توسط استرگاد^۲ یافت شد.

کرزیوکوسکی [۱۹] کاربردهایی از مسئله کلاه و انواع دیگر آن، و ارتباط آن‌ها با حیطه‌های مختلف علوم را توصیف می‌کند.

جانسون [۱۶] سخنرانی خود را با یک مسئله باز به پایان می‌رساند:

«یک سؤال طبیعی می‌تواند این باشد که؛ در حالتی که رنگ‌های کلاه دارای احتمالی یکسان نیستند، استراتژی بهینه چگونه تحت تأثیر قرار می‌گیرد؟»

در ادامه به پاسخ این سوال دست خواهید یافت. همچنین با روش ارائه‌شده به نتایج جالبی در مورد حالت متقارن خواهید رسید.

۲.۳ مجموعه‌های مناسب

۱.۲.۳ معرفی مجموعه‌های مناسب

ابتدا به بررسی بازی کلاه برای حالتی که ۳ بازیکن حضور دارد، می‌پردازیم. سپس در بخش بعدی مجموعه‌های مناسب در حالت کلی مورد تجزیه و تحلیل قرار می‌گیرند ($N > 1$). کلاه‌های سفید با احتمال p و کلاه‌های سیاه با احتمال q ($p + q = 1$) می‌باشند. بازیکنان با شماره‌های ۱، ۲ و ۳ برچسب‌گذاری می‌شوند. همچنین کلاه‌های سفید با برچسب ۰ و کلاه‌های سیاه با برچسب ۱، برچسب‌گذاری شده‌اند.

¹Östergård-Blass

²Östergård

در بازی کلاه، در حالتی که ۳ بازیکن حضور دارند، ۸ حالت متفاوت خواهید داشت:

جدول ۱.۳

احتمالات	S - کد	کد دودویی	ده‌دهی	$N = ۳$
$p^۳$	۰۰۰	۰۰۰	۰	
$p^۲q$	۱۱۰	۰۰۱	۱	
$p^۲q$	۲۰۱	۰۱۰	۲	
$pq^۲$	۳۱۱	۰۱۱	۳	
$p^۲q$	۰۲۲	۱۰۰	۴	
$pq^۲$	۱۳۲	۱۰۱	۵	
$pq^۲$	۲۲۳	۱۱۰	۶	
$q^۳$	۳۳۳	۱۱۱	۷	

S - کد نشان‌دهنده آن چیزی است که سه بازیکن مختلف مشاهده می‌کنند. با توجه به این که احتمال سفیدبودن کلاه‌ها p و سیاه‌بودن آن‌ها q است؛ لذا ستون پنجم جدول، نشان‌دهنده احتمال رخدادن کد متناظر در هر سطر آن است.

مثال ۱.۲.۳. فرض کنید برای بازیکنان ۱ - ۲ - ۳، قرارگرفتن رنگ کلاه‌ها به‌ترتیب، سفید - سیاه - سفید باشد. این ترتیب قرارگرفتن کلاه‌ها نشان‌دهنده کد دودویی ۰۱۰ می‌باشد؛ به این صورت که بازیکن اول کد ۱۰ را مشاهده می‌کند که در مبنای ده برابر ۲ است، بازیکن دوم کد ۰۰ را مشاهده می‌کند که در مبنای ده برابر ۰ و بازیکن سوم کد ۰۱ را مشاهده می‌کند که در مبنای ده برابر ۱ است؛ بنابراین S - کد متناظر با این حالت ۲۰۱ می‌باشد. کد ۰۱۰ در مبنای ده برابر ۲ است؛ در نتیجه همان‌طور که مشاهده می‌کنید مقدار ده‌دهی متناظر با کدها در ستون دوم نوشته شده است.

ملاحظه ۱.۲.۳. هر بازیکن دارای سه گزینه انتخابی «پاس = ۰»، «حدس سفید = ۱» و «حدس سیاه = -۱» می‌باشد.

تعریف ۱.۲.۳. ماتریس تصمیم $D = (a_{i,j})$ که یک ماتریس ۳×۴ است؛ به‌صورتی تعریف می‌شود که سطرهای آن با بازیکنان و ستون‌های آن با مقادیر S - کدهای ممکن برچسب‌گذاری شده‌اند؛ یعنی سطرهای آن با $i = ۱, ۲, ۳$ و ستون‌های آن با $j = ۰, ۱, ۲, ۳$ برچسب‌دهی می‌شوند. در این ماتریس مقدار $a_{i,j} \in \{-۱, ۰, ۱\}$ نشان‌دهنده این است که وقتی بازیکن i ، مقدار S - کد j را مشاهده می‌کند باید تصمیم $a_{i,j}$ را اتخاذ کند. (یادآوری می‌شود که مقدار ۰، ۱ و -۱ به‌ترتیب معادل تصمیم پاس، سفید و سیاه هستند).

با توجه به ماتریس تصمیم D ، احتمال برنده شدن (sum) را می توان از برنامه زیر محاسبه کرد:

sum=0

CASE 000 (three white hats) (Scode: 000)

If $a_{10} > -1$ and $a_{20} > -1$ and $a_{30} > -1$ and not ($a_{10} = a_{20} = a_{30} = 0$) then sum=sum+p³

CASE 001 (white,white,black) (Scode: 110)

If $a_{11} > -1$ and $a_{21} > -1$ and $a_{30} < 1$ and not ($a_{11} = a_{21} = a_{30} = 0$) then sum=sum+ p²q

CASE 010 (Scode: 201)

If $a_{12} > -1$ and $a_{20} < 1$ and $a_{31} > -1$ and not ($a_{12} = a_{20} = a_{31} = 0$) then sum=sum+p²q

CASE 011 (Scode: 311)

If $a_{13} > -1$ and $a_{21} < 1$ and $a_{31} < 1$ and not ($a_{13} = a_{21} = a_{31} = 0$) then sum=sum+pq²

CASE 100 (Scode: 022)

If $a_{10} < 1$ and $a_{22} > -1$ and $a_{32} > -1$ and not ($a_{10} = a_{22} = a_{32} = 0$) then sum=sum+p²q

CASE 101 (Scode: 132)

If $a_{11} < 1$ and $a_{23} > -1$ and $a_{32} < 1$ and not ($a_{11} = a_{23} = a_{32} = 0$) then sum=sum+pq²

CASE 110 (Scode: 223)

If $a_{12} < 1$ and $a_{22} < 1$ and $a_{33} > -1$ and not ($a_{12} = a_{22} = a_{33} = 0$) then sum=sum+pq²

CASE 111 (Scode: 333)

If $a_{13} < 1$ and $a_{23} < 1$ and $a_{33} < 1$ and not ($a_{13} = a_{23} = a_{33} = 0$) then sum=sum+q³

انتخاب های متفاوت برای D ، در واقع تعیین کننده این است که چه حالت هایی منجر به برنده شدن می شود؛ بنابراین این حالت ها در احتمال کلی مؤثر می باشند.

تعریف ۲.۲.۳. حالتی که منجر به برنده شدن می شود را حالت خوب و حالتی که باعث باخت می شود را حالت بد گوییم.

ملاحظه ۲.۲.۳. در کد نوشته شده فوق، برای هر حالتی که در آن $a_{i,j} > -1$ باشد یک حالت همتا وجود دارد که در آن $a_{i,j} < 1$ است و عکس این مطلب نیز برقرار می باشد.

مثال ۲.۲.۳. همتای $a_{۳۲} > -1$ برابر $a_{۳۲} < 1$ می باشد (بازیکن ۳ با S - کد ۲ و کدهای دودویی

۱۰۰ و ۱۰۱). می‌توان هم‌تا را با تغییر بیت^۱ وابسته به بیت بازیکن مناسب در کد دودویی پیدا کرد (در این مثال، بازیکن ۳ که بیت سوم کد دودویی ۱۰۰ است به ۱ تبدیل می‌شود و کد دودویی ۱۰۱ حاصل می‌شود).

نکته: وقتی یک عنصر $a_{i,j}$ و هم‌تایش، هر دو در حالت‌های خوب (حالتی که منجر به برد می‌شود) قرار دارند، باید $a_{i,j} = 0$ باشد.

توجه کنید هر حالت خوب حداقل یک $a_{i,j}$ دارد که مساوی صفر نیست؛ در این صورت هم‌تای خاص $a_{i,j}$ باید حداقل در یکی از حالت‌های بد باشد (برای این که هم‌تا نیز در یک حالت خوب قرار بگیرد، بایستی $a_{i,j} = 0$ باشد). اکنون آماده‌ی تعریف مجموعه مناسب می‌باشیم:

تعریف ۳.۲.۳. فرض کنید S مجموعه همه $S - A$ - کدها و A شامل تمام حالت‌های بد (با توجه به ماتریس تصمیم) باشد. هر عنصر در $S - A$ (حالت خوب) باید حداقل یک $a_{i,j}$ یکسان با هم‌تایش در A داشته باشد. مجموعه A با این ویژگی، مجموعه مناسب نامیده می‌شود؛ بنابراین داریم:

$$\forall s_1 s_2 s_3 \in S - A \exists a_1 a_2 a_3 \in A \exists i \in \{1, 2, 3\} : s_i = a_i.$$

یا به‌طور معادل:

$$\forall s_1 s_2 s_3 \in S \exists a_1 a_2 a_3 \in A \exists i \in \{1, 2, 3\} : s_i = a_i.$$

مثال ۳.۲.۳. فرض کنید $N = 3$ باشد. همان گونه که در جدول ۱.۳ مشاهده می‌کنید، مجموعه S برابر است با:

$$S = \{000, 110, 201, 311, 022, 132, 223, 333\}.$$

هدف، دستیابی به کمترین احتمال ممکن وضعیت‌های A می‌باشد (این که آیا با حالت خوب یا حالت بد سروکار دارید بستگی به انتخاب $a_{i,j}$ در ماتریس تصمیم D دارد. برای این منظور باید ماتریس تصمیم بهینه را پیدا کرد و مجموعه مناسب اولین و مهم‌ترین مرحله می‌باشد).

واضح است که هر مجموعه مناسب در S (مجموعه $S - A$ - کدها) دارای کد پوششی دودویی از شعاع ۲ است.

نمادگذاری: از نماد AS به‌عنوان مجموعه‌ای از تمام مجموعه‌های مناسب استفاده می‌شود.

¹Flipping

بخش ۴.۳ نشان می‌دهد که وقتی $N = 3$ است؛ آن گاه $AS = \{\{0, 7\}, \{1, 6\}, \{2, 5\}, \{3, 4\}\}$ می‌باشد، که در مجموعه AS از نمایش ددهی به جای نمایش دودویی استفاده می‌شود. همچنین در این بخش ثابت می‌شود زمانی که $p = 0/5$ باشد؛ آن گاه همه ۴ عناصر AS بهینه هستند و در حالت $p \neq 0/5$ ، تنها سه عنصر آخر بهینه می‌باشد. برای زمانی که $N = 3$ است، ۴ ماتریس تصمیم ساخته می‌شود:

$$\{0, 7\} \in AS$$

کد دودویی: ۰۰۰ و ۱۱۱

بیت‌های کد دودویی، مقدار (ارزش) در ماتریس تصمیم را به صورت زیر تعیین می‌کنند:

– بیت ۰ در کد دودویی نشانگر رنگ سفید و بیت ۱ نشانگر رنگ سیاه می‌باشد. زمانی که کد دودویی در مجموعه مناسب دارای بیت ۰ است؛ در این صورت چون همتای کد در حالت خوب قرار دارد، دارای بیت ۱ می‌باشد که مطابق با حدس – کد «۱-» (حدس سیاه) است.

– اگر کد دودویی دارای بیت ۱ باشد؛ آن گاه بیت همتای کد، مقدار ۰ می‌گیرد و منجر به حدس – کد «۱» (حدس سفید) می‌شود.

نتیجه ۱.۲.۳. مقدار – حدس در ماتریس تصمیم توسط بیت – تابع $2b - 1 \rightarrow b$ تعیین می‌شود.

S – کد: ۰۰۰ و ۳۳۳

(با توجه به مقدار متناظر با ستون‌ها، ماتریس تصمیم به صورت یکتا تعیین می‌شود).

ماتریس تصمیم: $\{1, 6\} \in AS$

	S – کد			
	۰	۱	۲	۳
بازیکن ۱	-۱	۰	۰	۱
بازیکن ۲	-۱	۰	۰	۱
بازیکن ۳	-۱	۰	۰	۱

کد دودویی: ۰۰۱ و ۱۱۰

S – کد: ۱۱۰ و ۲۲۳

ماتریس تصمیم:

	S – کد			
	۰	-۱	۱	۰
بازیکن ۱	۰	-۱	۱	۰
بازیکن ۲	۰	-۱	۱	۰
بازیکن ۳	۱	۰	۰	-۱

$$\{2, 5\} \in \mathcal{AS}$$

کد دودویی: ۰۱۰ و ۱۰۱

S - کد: ۱۳۲ و ۲۰۱

ماتریس تصمیم:

	S - کد			
بازیکن ۱	۰	۱	-۱	۰
بازیکن ۲	۱	۰	۰	-۱
بازیکن ۳	۰	-۱	۱	۰

$$\{3, 4\} \in \mathcal{AS}$$

کد دودویی: ۱۰۰, ۰۱۱

Scode: ۰۲۲ و ۳۱۱

ماتریس تصمیم:

	S - کد			
بازیکن ۱	۱	۰	۰	-۱
بازیکن ۲	۰	۱	-۱	۰
بازیکن ۳	۰	۱	-۱	۰

اولین ماتریس تصمیم مربوط به یک استراتژی شناخته شده برای زمانی که $p = ۰/۵$ است، می باشد. این استراتژی عبارت است از: «زمانی که دو بازیکن دیگر دارای رنگ های مختلف هستند، پاس دهید؛ در غیر این صورت رنگ مخالف را انتخاب کنید.»
شانس برنده شدن این استراتژی $\frac{3}{4}$ می باشد و این یک استراتژی بهینه برای $N = 3$ است. وقتی $p \neq ۰/۵$ باشد، این استراتژی بهینه نخواهد بود. (بخش ۲.۴.۳ را مشاهده کنید).

۲.۲.۳ مجموعه های مناسب در حالت کلی ($N > 1$)

از آنجا که N بازیکن شرکت کننده در بازی، افراد متفاوت هستند؛ بنابراین می توان آن ها را از ۱ تا N برچسب گذاری کرد. هر حالت ممکن از کلاه سفید (کد ۰) و سیاه (کد ۱) توسط یک عنصر از $B = \{b_1 b_2 \dots b_N \mid b_i \in \{0, 1\}, i = 1, 2, \dots, N\}$ قابل نمایش می باشد.

تعریف ۴.۲.۳. S - کد متناظر با هر بازیکن نشان دهنده آن چیزی است که بازیکن مشاهده می کند. در واقع بازیکن i ، کد دودویی $b_1 \dots b_{i-1} b_{i+1} \dots b_N$ را با ارزش دهدهی

$s_i = \sum_{k=1}^{i-1} b_k \cdot 2^{N-k-1} + \sum_{k=i+1}^N b_k \cdot 2^{N-K}$ که مقداری بین ۰ و $2^{N-1} - 1$ است، را می بیند.

تعریف ۵.۲.۳. فرض کنید S مجموعه همه S - کدها به صورت:

$$S = \{s_1 s_2 \dots s_N \mid s_i = \sum_{k=1}^{i-1} b_k \cdot 2^{N-k-1} + \sum_{k=i+1}^N b_k \cdot 2^{N-K}, b_i \in \{0, 1\}, i = 1, 2, \dots, N\}$$

و A زیرمجموعه ای ناتهی از S باشد. مجموعه A برای S مناسب است اگر،

$$\forall s_1 s_2 \dots s_N \in S \exists a_1 a_2 \dots a_N \in A \exists i \in \{1, 2, \dots, N\} : s_i = a_i.$$

توجه کنید که چنین مجموعه ای همیشه وجود دارد؛ زیرا S برای خودش همواره مناسب است. به عبارت دیگر، مجموعه مناسب در S (مجموعه S - کدها)، دارای کدهای پوششی دودویی از شعاع $N - 1$ می باشد.

یادآوری: هر بازیکن دارای سه گزینه انتخابی «پاس = ۰»، «حدس سفید = ۱» و «حدس سیاه = -۱» می باشد.

تعریف ۶.۲.۳. ماتریس تصمیم $D = (a_{i,j})$ یک ماتریس $1 - 2^{N-1} \times N$ بُعدی است که در آن $a_{i,j} \in \{-1, 0, 1\}$ و سطرهای آن متناظر با بازیکنان و ستون های آن متناظر با S - کدها می باشد.

مقدار $a_{i,j}$ نشان دهنده این است که وقتی بازیکن i ، S - کد j را مشاهده می کند باید تصمیم $a_{i,j}$ را اخذ کند (با توجه به مقدار $a_{i,j}$ یکی از حدس های پاس، سیاه یا حدس سفید را می زند).

در ادامه، مجموع احتمالات (sum) حدس ها مشاهده می شود:

برای یک حالت دلخواه $b_1 b_2 \dots b_N$ در B با n صفر (تعداد b_i هایی که برابر صفر هستند و

$$sum = 0, (n = 0, 1, \dots, N)$$

احتمال رخ دادن حالت $b_1 \dots b_N$ (S - کد بازیکن i : $s_i = \sum_{k=1}^{i-1} b_k \cdot 2^{N-k-1} + \sum_{k=i+1}^N b_k \cdot 2^{N-K}$) اگر $a_{1,s_1} \in \{0, 1 - 2b_1\}$ و $a_{2,s_2} \in \{0, 1 - 2b_2\}$ و ... و $a_{N,s_N} \in \{0, 1 - 2b_N\}$ باشد و همچنین همه a_{i,s_i} ها ($i = 1, \dots, N$) برابر با صفر نباشد؛ در این صورت $sum = sum + p^n q^{N-n}$ می باشد.

در ادامه با توجه به ماتریس D ، مقدار احتمال موفقیت گروه محاسبه می شود. برای هر

حالت دلخواه $b_1 b_2 \dots b_N$ بازیکن i - ام مقدار S - کد $s_i = \sum_{k=1}^{i-1} b_k \cdot 2^{N-k-1} + \sum_{k=i+1}^N b_k \cdot 2^{N-K}$ را مشاهده می کند؛ بنابراین براساس ماتریس D ، تصمیم متناظر با $a_{i,j}$ را اخذ می کند. زمانی این تصمیم منجر به برد می شود که،

۱. برای هر $i (i = 1, 2, \dots, N)$ ، $a_{i,s_i} \in \{0, 1 - 2b_i\}$ باشد و

۲. حداقل یکی از a_{i,s_i} ها برای $i = 1, 2, \dots, N$ مخالف صفر باشد.

بنابراین الگوریتم زیر مقدار احتمال موفقیت گروه را محاسبه می‌کند.

$$sum = \circ \quad \bullet$$

- برای حالت دلخواه $b_1 b_2 \dots b_N \in B$ با n درایه صفر، اگر $b_1 b_2 \dots b_N$ در موارد (۱) و (۲) صدق کند؛ آن‌گاه $sum = sum + p^n q^{N-n}$ می‌باشد.

هر انتخاب $a_{i,j}$ در ماتریس تصمیم تعیین می‌کند که کدام حالت‌ها در احتمال کلی مؤثر هستند؛ یعنی منجر به برد می‌شوند (حالت خوب) و کدام حالت‌ها در مجموع تأثیر نمی‌گذارد؛ یعنی منجر به باخت می‌شوند (حالت بد).

ملاحظه ۳.۲.۳. اگر حالتی منجر به برد شود، باعث تغییر مجموع (sum) می‌شود؛ در غیر این صورت مجموع بدون تغییر باقی می‌ماند.

حال روی بازیکن i تمرکز می‌کنیم. ($i = 1, 2, \dots, N$)

هر $a_{i,j} \in \{\circ, -1\}$ یک همتای $a_{i,j} \in \{\circ, -1\}$ دارد و برعکس. با استفاده از روش تغییر بیت در موقعیت i داریم:

حالت $b_1 \dots b_i \dots b_N$ به حالت $b_1 \dots b_i - 1 \dots b_N$ و $b_1 \dots b_i - 1 \dots b_N$ به $b_1 \dots b_i \dots b_N$ تبدیل می‌شود. این بدان معنی است که اگر هر دو حالت در یک راه‌حل از بازی حضور داشته باشند، $a_{i,j} = \circ$ می‌باشد.

هر حالت خوب حداقل یک $a_{i,j}$ دارد که مساوی صفر نیست. همتای خاص $a_{i,j}$ ، باید در حالت بد باشد؛ در غیر این صورت $a_{i,j} = \circ$ می‌باشد.

فرض کنید A زیرمجموعه S با همه حالت‌های بد باشد. اکنون به شرح زیر است:

$$\forall s_1 s_2 \dots s_N \in S-A \quad \exists a_1 a_2 \dots a_N \in A \quad \exists i \in \{1, 2, \dots, N\} : s_i = a_i$$

و این معادل با تعریف مجموعه مناسب است.

با توجه به تعریف مجموعه مناسب، یک ماتریس تصمیم‌گیری D با روش زیر ساخته می‌شود. در پیوست ب یک ماتریس مولد تصمیم‌گیری با استفاده از گام‌های زیر طراحی شده است:

۱. ابتدا تمامی عناصر D را صفر (= پاس) قرار دهید.

۲. سپس برای هر عنصر در مجموعه مناسب:

الف) تعیین نمایش دودویی (bincode) $b_1 b_2 \dots b_N$.

ب) محاسبه S - کدها: $s_i = \sum_{k=1}^{i-1} b_k \cdot 2^{N-k-1} + \sum_{k=i+1}^N b_k \cdot 2^{N-k}$ ($i = 1, 2, \dots, N$)

ج) برای هر بازیکن i : $a_{i,s_i} = 2b_i - 1$ ($i = 1, 2, \dots, N$) (بیت-تابع $b \rightarrow 2b - 1$ ، در بخش ۱.۲.۳ مشاهده کنید).

در بخش ۵.۳ زمانی که $N = 4$ است، یک مجموعه مناسب با 4^0 عنصر موجود می‌باشد که هر عنصر دارای بُعد ۴ است.

نمادگذاری: از نماد nas برای تعداد عناصر در یک مجموعه مناسب ($N = 4 : nas = 4^0$)، و از نماد das برای بُعد هر عنصر مجموعه مناسب ($N = 4 : das = 4$) استفاده می‌شود.

دقت کنید تعریف مجموعه مناسب به p بستگی ندارد؛ بنابراین nas و das مستقل از p می‌باشند. همان‌طور که پیش‌تر گفته شد، وقتی $N = 4$ است، 4^0 مجموعه مناسب داریم که دارای بُعد ۴ می‌باشد. حال دو مجموعه $\{1, 6, 10, 13\}$ و $\{1, 6, 9, 14\}$ از این 4^0 مجموعه انتخاب می‌شوند. ابتدا مجموعه ۴ عنصری $\{1, 6, 10, 13\}$ را در نظر بگیرید:

S - کد	کد دودویی
۱۱۱۰	۰۰۰۱
۶۲۲۳	۰۱۱۰
۲۶۴۵	۱۰۱۰
۵۵۷۶	۱۱۰۱

S - کد: موقعیت (ستون) در ماتریس
 کد دودویی: مقدار بیت - تابع $1 - 2b \rightarrow b$.
 ماتریس تصمیم:

	S - کد:	۰	۱	۲	۳	۴	۵	۶	۷
بازیکن ۱	۰	۰	-۱	۱	۰	۰	۱	-۱	۰
بازیکن ۲	۰	۰	-۱	۱	۰	۰	۱	-۱	۰
بازیکن ۳	۰	۰	-۱	۱	۰	۱	۰	۰	-۱
بازیکن ۴	۱	۰	۰	۰	-۱	۰	-۱	۱	۰

حال مجموعه $\{1, 6, 9, 14\}$ مورد بررسی قرار می‌گیرد:

	کد دودویی	S -کد
۱	۰۰۰۱	۱۱۱۰
۶	۰۱۱۰	۶۲۲۳
۹	۱۰۰۱	۱۵۵۴
۱۴	۱۱۱۰	۶۶۶۷

ماتریس تصمیم:

	S -کد:							
	۰	۱	۲	۳	۴	۵	۶	۷
بازیکن ۱	۰	*	۰	۰	۰	۰	*	۰
بازیکن ۲	۰	-۱	۱	۰	۰	-۱	۱	۰
بازیکن ۳	۰	-۱	۱	۰	۰	-۱	۱	۰
بازیکن ۴	۱	۰	۰	-۱	۱	۰	۰	-۱

بازیکن و S -کد، سطر و ستون مربوطه را تعیین می‌کنند. در موقعیت $(1, 1)$ توسط بیت-تابع $b \rightarrow 2b - 1$ ، مقادیر -1 و 1 را داریم (منطقه سایه‌دار در S -کد و کد دودویی را مشاهده کنید). به نظر می‌رسد هیچ راه‌حلی وجود ندارد؛ اما مجموعه مناسب شامل هر دو کد 1 و 9 نیز می‌باشد که در مجموع احتمال محاسبه نخواهند شد؛ لذا بازیکن 1 با S -کد 1 و کد دودویی هر دو 0 و 1 ، به این معنی است که می‌توان دو محدودیت $a_{i,j} > -1$ و $a_{i,j} < 1$ را حذف کرد؛ بنابراین آزادی کامل داریم. همان داستان در موقعیت $(1, 6)$ نیز رخ می‌دهد. در ماتریس D از علامت $*/$ استفاده می‌شود؛ به این معنی که هر یک از گزینه‌های 0 ، 1 و -1 مجاز است.

۳.۳ بازی کلاه دونفره

در ابتدا، بازی کلاه دونفره به نظر می‌رسد بدیهی باشد؛ اما رویکرد مجموعه مناسب، تصویر دیگری را ارائه می‌دهد.

۱.۳.۳ بازی کلاه دونفره متقارن

یک مجموعه مناسب می‌تواند با $\{i_1, i_2, \dots, i_{das}\}$ توضیح داده شود که در آن $0 \leq i_1 < i_2 < \dots < i_{das} \leq 2^N - 1$ قرار دارد و das ، بُعد مجموعه مناسب است. پیوست آ، شامل یک برنامه VBA-Excel می‌باشد که می‌تواند برای پیدا کردن تمام مجموعه‌های مناسب مورد استفاده قرار گیرد (p, N و das برای انتخاب آزاد هستند). این برنامه یک آزمون سعی و خطا روی S - کد مجموعه S می‌باشد و چک می‌کند که آیا انتخاب $\{i_1, i_2, \dots, i_{das}\}$ ، یک مجموعه مناسب را ارائه می‌دهد؟

در این حالت به‌طور مستقیم از تعریف مجموعه مناسب استفاده می‌شود. بُعد مجموعه مناسب (das) را نمی‌دانیم؛ بنابراین سعی می‌شود ابتدا برنامه را با $N = 2$ ، $p = 0.5$ و $das = 1$ اجرا کنیم، هیچ مجموعه مناسبی پیدا نمی‌شود و لذا $das = 2$ را انتخاب می‌کنیم و نتیجه زیر حاصل می‌شود (تمام محاسبات انجام‌شده در این پایان‌نامه بر روی یک لپ‌تاپ استاندارد ساخته شده است):

i_1	i_2	مجموع	تعداد ($0, bin.i_1$)	تعداد ($0, bin.i_2$)
0	1	0.5	2	1
0	2	0.5	2	1
0	3	0.5	2	0
1	2	0.5	1	1
1	3	0.5	1	0
2	3	0.5	1	0

مجموعه AS به‌صورت $AS = \{\{0, 1\}, \{0, 2\}, \{0, 3\}, \{1, 2\}, \{1, 3\}, \{2, 3\}\}$ می‌باشد؛ بنابراین $das = 2$ و $nas = 6$ می‌باشد. هر عنصر در مجموعه AS دارای احتمال $p = 0.5$ است.

توجه: دو ستون آخر نشان‌دهنده تعداد صفر در کد دودویی i_1 و i_2 است که نقش مهمی را در حالت نامتقارن بازی می‌کنند.

اگر $p = 0.5$ باشد، مجموعه مناسب با کوچک‌ترین das مجموعه‌ای بهینه است. اگر $das = 3$ باشد، مجموع برابر 0.75 و اگر $das = 4$ باشد، مجموع در حالت‌های بد (مجموعه مناسب)، ۱ است؛ بنابراین احتمالات برنده 0.25 و 0 می‌باشد. در نتیجه اگر $N = 2$ و $p = 0.5$ باشد؛ آن‌گاه $nas = 6$ و هر عنصر بهینه است؛ لذا $nasopt = 6$ می‌باشد.

نمادگذاری: از نماد $nasopt$ به‌عنوان تعداد عناصر در مجموعه بهینه مطلوب استفاده می‌شود.

روش بخش ۲.۳، ماتریس تصمیم زیر را ارائه می‌دهد:

$\{\circ, 1\}$	$\{\circ, 2\}$	$\{\circ, 3\}$	$\{1, 2\}$	$\{1, 3\}$	$\{2, 3\}$
-1 -1	$*$ $\circ$	-1 1	1 -1	$\circ$ $*$	1 1
$*$ $\circ$	-1 -1	-1 1	1 -1	1 1	$\circ$ $*$

با اجرای برنامه در پیوست ب نیز همین نتیجه حاصل می‌شود. یک ماتریس مولد با پارامترهای $N = 2$ ، $p = \circ/5$ و $nasopt = 6$ دریافت می‌شود (ماتریس مولد، کد ۳ را به جای * نشان می‌دهد).

با روش بخش ۲.۳ نیز ماتریس‌هایی می‌توان به دست آورد (عیناً ماتریس‌های ماتریس مولد را ارائه می‌دهد) اما همه ماتریس‌ها را ارائه نمی‌دهد.

اگر مایل به دریافت تمام ماتریس‌های بهینه باشید، بایستی با ماتریس تصمیم $\begin{pmatrix} a_{1\circ} & a_{11} \\ a_{2\circ} & a_{21} \end{pmatrix}$ شروع کنید.

ابتدا مجموعه $\{\circ, 3\} \in AS$ را در نظر بگیرید:

bin.code=00 and 11; Scode 00 and 11.

sum=0

CASE bin.code=00 (Scode: 00)

If $a_{10} > -1$ $a_{20} > -1$ and not ($a_{10} = a_{20} = 0$) then sum=sum+p²

CASE bin.code=11 (Scode: 11)

If $a_{11} < 1$ and $a_{21} < 1$ and not ($a_{11} = a_{21} = 0$) then sum=sum+p²

در این حالت، $a_{i,j}$ ‌های مورد بحث در الگوریتم بالا در دو قسمت اشتراکی ندارند؛ لذا $2^2 - 1$ ماتریس در هر حالت داریم، در نتیجه کل ماتریس‌ها به تعداد $9 = (2^2 - 1)^2$ می‌باشد. به روش مشابه، مجموعه $\{1, 2\}$ نیز ۹ ماتریس می‌دهد.

اما برای مجموعه $\{\circ, 1\}$ ، درایه $a_{2\circ}$ در هر دو قسمت الگوریتم حضور دارد و لذا تعداد ماتریس‌ها کمتر خواهد بود.

Bin.code=00 and 01; Scode 00 and 10.

sum=0

CASE bin.code=00 (Scode: 00)

If $a_{10} > -1$ $a_{20} > -1$ and not ($a_{10} = a_{20} = 0$) then sum=sum+p²

CASE bin.code=01 (Scode: 10)

If $a_{11} > -1$ and $a_{20} < 1$ and not ($a_{11} = a_{20} = 0$) then sum=sum+pq

اکنون a_2 باید صفر باشد و $3 = 1 - 2^2$ ماتریس داریم.

در کل، به تعداد $3^0 = 2^9 + 4^3$ ماتریس داریم؛ در حالی که 1^3 ماتریس هم‌ریخت (متناظر) می‌باشند. تفاوت بین دو ماتریس، تعویض بازیکنان ۱ و ۲ است. 1^7 ماتریس ناهم‌ریخت عبارتند از:

$\begin{pmatrix} -1 & -1 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & -1 \\ 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & -1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & 0 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & 0 \\ -1 & 1 \end{pmatrix}$	$\begin{pmatrix} -1 & 1 \\ -1 & 1 \end{pmatrix}$	$\begin{pmatrix} -1 & 1 \\ 0 & 0 \end{pmatrix}$	$\begin{pmatrix} -1 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & -1 \\ 1 & -1 \end{pmatrix}$
$\begin{pmatrix} 0 & -1 \\ 1 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & -1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}$	$\begin{pmatrix} 0 & 0 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 \\ 1 & -1 \end{pmatrix}$	$\begin{pmatrix} 1 & -1 \\ 1 & 0 \end{pmatrix}$	

همه 3^0 ماتریس نیز توسط اجرای برنامه در پیوست پ به دست می‌آید. استفاده از روش سعی و خطا برای $N = 2$ امکان‌پذیر است.

۲.۳.۳ بازی کلاه دونفره نامتقارن

در بازی‌های نامتقارن کلاه (دو یا چند نفره)، $p > 0.5$ انتخاب می‌شود. این انتخاب هیچ محدودیتی را ایجاد نمی‌کند؛ زیرا می‌توان نقش p و q را زمانی که $p < 0.5$ است را معکوس کرد.

الگوریتم مجموعه مولد از پیوست آ، با پارامترهای $N = 2$ ، $p = 0.9$ و $das = 1$ اجرا می‌شود. در این حالت هیچ مجموعه مناسبی وجود ندارد. برای $das = 2$ شرایط زیر را داریم:

i_1	i_2	مجموع	تعداد $(0, bin.i_1)$	تعداد $(0, bin.i_2)$
۰	۱	۰/۹	۲	۱
۰	۲	۰/۹	۲	۱
۰	۳	۰/۸۲	۲	۰
۱	۲	۰/۱۸	۱	۱
۱	۳	۰/۱	۱	۰
۲	۳	۰/۱	۱	۰

با توجه به تعریف مجموعه مناسب، می‌دانیم که مجموعه مناسب مستقل از p می‌باشد. ستون مجموع را می‌توان به‌طور کلی توسط رابطه $p^{c_1} q^{2-c_1} + p^{c_2} q^{2-c_2}$ به دست آورد، جایی که؛ $c_1 = \text{تعداد}(0, bin.i_1)$ و $c_2 = \text{تعداد}(0, bin.i_2)$ می‌باشد:

i_1	i_2	مجموع	تعداد ($\circ, bin.i_1$)	تعداد ($\circ, bin.i_2$)
$\circ$	۱	$p^2 + pq$	۲	۱
$\circ$	۲	$p^2 + pq$	۲	۱
$\circ$	۳	$p^2 + q^2$	۲	$\circ$
۱	۲	$2pq$	۱	۱
۱	۳	$q^2 + pq$	۱	$\circ$
۲	۳	$q^2 + pq$	۱	$\circ$

اگر $p > q$ باشد، $q^2 + pq$ کوچک‌ترین مجموع احتمالات می‌باشد؛ بنابراین دو مجموعه مناسب بهینه داریم: $\{1, 3\}$ و $\{2, 3\}$.

تعریف ۱.۳.۳. فرض کنید $\Psi(N, p)$ ماکزیمم احتمال حدس درست در بازی کلاه باشد که در آن N نشان‌دهنده تعداد بازیکنان و p برابر با احتمال گرفتن کلاه سفید است.

توجه ۱.۳.۳.

$$\Psi(2, p) = 1 - (pq + q^2) = 1 - q(p + q) = 1 - q = p.$$

همان گونه که انتظار داشتیم $\Psi(2, p) = p$ می‌باشد.

نتیجه ۱.۳.۳. همه ماتریس‌های تصمیم بهینه عبارتند از:

$\{1, 3\}$	$\{2, 3\}$
$\circ$ *	۱ ۱
۱ ۱	$\circ$ *

همان‌طور که قبلاً این نتیجه با استفاده از ماتریس تصمیم مولد پیوست ب با $das = 2, p > 0.5$ و $nasopt = 6$ به دست آمد.

ملاحظه ۱.۳.۳. $das = 2$ اولین راه‌حل مناسب را ارائه می‌دهد؛ اما برای $das > 2$ چگونه عمل می‌کنند؟

زمانی که $p = 0.5$ است مشکلی وجود ندارد؛ اما برای $p > 0.5$ باید هوشیار بود. اجرای برنامه پیوست آ، با مقادیر مختلف das نشان می‌دهد که $das = 2$ بهینه است:

تعداد $\circ$	$\circ$	۱	۲
عبارت مرتبط	q^2	pq	p^2
$das = 2$	۱	۱	$\circ$
$das = 3$	۱	۲	$\circ$
$das = 4$	۱	۲	۱

۴.۳ بازی کلاه سه نفره

۱.۴.۳ بازی کلاه سه نفره متقارن

مجموعه مناسب مولد در پیوست آ اجرا می شود (p, N) و das برای انتخاب آزاد هستند). از آنجا که بعد مینیمال مجموعه مناسب را نمی دانیم؛ بنابراین سعی می کنیم ابتدا برنامه را با $p = \circ/5$ ، $N = 3$ و $das = 1$ اجرا کنیم. هیچ مجموعه مناسبی پیدا نمی شود. بعد $das = 2$ را امتحان می کنیم و نتیجه زیر حاصل می شود:

i_1	i_2	مجموع	تعداد $(\circ, bin.i_1)$	تعداد $(\circ, bin.i_2)$
$\circ$	۷	$\circ/25$	۳	$\circ$
۱	۶	$\circ/25$	۲	۱
۲	۵	$\circ/25$	۲	۱
۳	۴	$\circ/25$	۱	۲

مجموعه AS به صورت $AS = \{\{\circ, 7\}, \{1, 6\}, \{2, 5\}, \{3, 4\}\}$ می باشد؛ بنابراین وقتی $das = 2$ است، مجموعه AS دارای ۴ عنصر می باشد ($nas = 4$)، که احتمال هر کدام $\circ/25$ است.

توجه: دو ستون آخر تعداد صفرها در کد دودویی i_1 و i_2 را نشان می دهند.

اگر $p = \circ/5$ باشد؛ آن گاه مجموعه مناسب با das مینیمال نیز مجموعه بهینه است. مجموع احتمالات در یک مجموعه مناسب برابر با $(das = 2, 3, \dots, 8)$ می باشد. در حالت $N = 3$ با پیروی از روش بخش ۲.۳، دقیقاً یک ماتریس تصمیم برای هر عنصر AS پیدا می شود:

$\{0, 7\}$	$\{1, 6\}$	$\{2, 5\}$	$\{3, 4\}$
کد دودویی: ۰۰۰, ۱۱۱	کد دودویی: ۰۰۱, ۱۱۰	کد دودویی: ۰۱۰, ۱۰۱	کد دودویی: ۰۱۱, ۱۰۰
$-S: ۰۰۰, ۳۳۳$	$-S: ۱۱۰, ۲۲۳$	$-S: ۲۰۱, ۱۳۲$	$-S: ۳۱۱, ۰۲۲$
$\begin{matrix} -1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 1 \\ -1 & 0 & 0 & 1 \end{matrix}$	$\begin{matrix} 0 & -1 & 1 & 0 \\ 0 & -1 & 1 & 0 \\ 1 & 0 & 0 & -1 \end{matrix}$	$\begin{matrix} 0 & 1 & -1 & 0 \\ 1 & 0 & 0 & -1 \\ 0 & -1 & 1 & 0 \end{matrix}$	$\begin{matrix} 1 & 0 & 0 & -1 \\ 0 & 1 & -1 & 0 \\ 0 & 1 & -1 & 0 \end{matrix}$

توجه کنید که نتایج یکسان توسط اجرای ماتریس مولد پیوست ب، با پارامترهای $N = 3$ ، $nas = nasopt = 4$ و $das = 2$ به دست می‌آید. همچنین با توجه به پیوست ت، می‌توان نتیجه گرفت که ماتریس‌های بهینه بیش‌تری وجود ندارد.

۲.۴.۳ بازی کلاه سه نفره نامتقارن

از مجموعه مناسب مولد پیوست آ، با پارامترهای $N = 3$ ، و $p = 0/9$ استفاده می‌کنیم. تنظیم $das = 2$ مجموعه‌های مناسب زیر را ارائه می‌دهد:

i_1	i_2	مجموع	تعداد $(0, bin.i_1)$	تعداد $(0, bin.i_2)$
۰	۷	$0/73$	۳	۰
۱	۶	$0/09$	۲	۱
۱	۵	$0/09$	۲	۲
۲	۴	$0/09$	۱	۳

سه مجموعه بهینه مشاهده می‌شود: $\min \text{sum} = 0/09$ ساختن مجموعه مناسب با استفاده از تعریف، مستقل از پارامتر p می‌باشد. بنابراین به‌طور کلی دریافت می‌کنیم:

i_1	i_2	مجموع	تعداد $(0, bin.i_1)$	تعداد $(0, bin.i_2)$
۰	۷	$p^3 + q^3$	۳	۰
۱	۶	$p^2q + pq^2$	۲	۱
۱	۵	$p^2q + pq^2$	۲	۲
۲	۴	$p^2q + pq^2$	۱	۳

اگر $p > q$ باشد؛ آن‌گاه $p^3 + q^3 > p^2q + pq^2$ می‌باشد و لذا مجموعه $\{0, 7\}$ بهینه نیست.

فرض کنید $\Psi(N, p)$ ماکزیمم احتمال حدس درست در بازی باشد.

$$\begin{aligned}\Psi(3, p) &= 1 - (pq^2 + p^2q) = 1 - pq(q + p) = 1 - pq = 1 - p + p^2 \\ &= \frac{3}{4} + \frac{1}{4} - p + p^2 = \frac{3}{4} + (p - \frac{1}{4})^2 = 0.75 + (p - 0.25)^2.\end{aligned}$$

ماتریس‌های تصمیم مربوط را می‌توان در حالت $p = 0.5$ با $AS = \{\{1, 6\}, \{2, 5\}, \{3, 4\}\}$ به دست آورد.

ملاحظه ۱.۴.۳. $das = 2$ اولین مجموعه مناسب را ارائه می‌دهد؛ اما برای $das > 2$ چگونه عمل می‌کنند؟

نتیجه مشابهی با اجرای برنامه آزمون سعی و خطا در پیوست ت، با هر $p \neq 0.5$ حاصل می‌شود.

در حالت $p = 0.5$ مشکلی وجود ندارد؛ اما زمانی که $p > 0.5$ باشد، مطمئن نیستیم. اجرای مجموعه مناسب مولد از پیوست آ، با مقادیر مختلف das ، این نتیجه را می‌دهد که $das = 2$ بهینه‌ترین است (به دنبال کوتاه‌ترین راه حل هستیم).

تعداد ۰	۰	۱	۲	۳
عبارت مرتبط	q^3	q^2p	qp^2	p^3
$das = 2$	۰	۱	۱	۰
$das = 3$	۱	۱	۱	۰
$das = 4$	۱	۲	۱	۰
$das = 5$	۱	۳	۱	۰
$das = 6$	۱	۳	۲	۰
$das = 7$	۱	۳	۳	۰
$das = 8$	۱	۳	۳	۱

۵.۳ بازی کلاه چهارنفره

۱.۵.۳ بازی کلاه چهارنفره متقارن

مجموعه AS مولد در پیوست آ اجرا می‌شود. ابعاد یک مجموعه مناسب مینیمال را نمی‌دانیم؛ بنابراین سعی می‌کنیم ابتدا برنامه را با $N = 4$ ، $p = 0.5$ ، $das = 2$ و ۳ اجرا کنیم؛ در این صورت $AS = \phi$ می‌باشد. سپس $das = 4$ را امتحان می‌کنیم و به عنوان نتیجه، یک لیست از ۴۰

مجموعه مناسب، هر کدام با احتمال $p = 0.25$ را دریافت می‌کنیم.
ده مجموعه مناسب از آن لیست ارائه می‌شود:

i_1	i_2	i_3	i_4	مجموع	$0, bin.i_1$	$0, bin.i_2$	$0, bin.i_3$	$0, bin.i_4$
۰	۱	۱۴	۱۵	۰.۲۵	۴	۳	۱	۰
۰	۲	۱۳	۱۵	۰.۲۵	۴	۳	۱	۰
۰	۳	۱۳	۱۴	۰.۲۵	۴	۲	۱	۱
۰	۴	۱۱	۱۵	۰.۲۵	۴	۳	۱	۰
۰	۵	۱۱	۱۴	۰.۲۵	۴	۲	۱	۱
۰	۶	۱۱	۱۳	۰.۲۵	۴	۲	۱	۱
۰	۷	۸	۱۵	۰.۲۵	۴	۱	۳	۰
۰	۷	۹	۱۴	۰.۲۵	۴	۱	۲	۱
۰	۷	۱۰	۱۳	۰.۲۵	۴	۱	۲	۱
۰	۷	۱۱	۱۲	۰.۲۵	۴	۱	۱	۲

احتمال مطلوب بازی $0.75 = 1 - 0.25$ است. اگر $p = 0.5$ باشد، مجموعه مناسب با کمترین das نیز مجموعه بهینه می‌باشد. مجموع احتمالات در مجموعه مناسب $\frac{das}{44}$ ($das = 4, 5, \dots, 16$) است. در حالت $N = 4$ با پیروی از بخش ۲.۳، ماتریس تصمیم برای هر یک از ۴۰ مجموعه مناسب پیدا می‌شود.
نتیجه با استفاده از ماتریس مولد پیوست ب، دریافت می‌شود. نتیجه اول را مشاهده کنید:
(در خط اول چهارعنصر سایه‌دار، یک مجموعه مناسب را نشان می‌دهد).

					۰	۱	۱۴	۱۵	
-۱	-۱	۰	۰	۰	۰	۰	۰	۱	۱
-۱	-۱	۰	۰	۰	۰	۰	۰	۱	۱
-۱	-۱	۰	۰	۰	۰	۰	۰	۱	۱
۳	۰	۰	۰	۰	۰	۰	۰	۰	۳

توجه: بازیکنان ۱ - ۲ - ۳ مانند $N = 3$ رفتار می‌کنند، بازیکن ۴ می‌تواند همیشه پاس دهد.
پروژه تمدیدشده در بخش ۱.۳.۳، ماتریس تصمیم افزوده را می‌دهد.

۲.۵.۳ بازی کلاه چهارنفره نامتقارن

برنامه VBA پیوست آ، با پارامترهای $N = 4$ و $p = 0.9$ اجرا می‌شود. تنظیم $das = 4$ ، ۴۰ مجموعه مناسب را تولید می‌کند (پیوست ث برای یک لیست مرتب‌شده بر اساس مجموع را

مشاهده کنید). کمترین مجموع $p = 0/09$ است و ۲۴ مجموعه مناسب بهینه می باشد. با توجه به تعریف، ساخت یک مجموعه مناسب مستقل از p است و ستون احتمال فقط تعداد صفر را در هر عنصر (دودویی) یک مجموعه مناسب محاسبه می کند. زمانی که تعداد صفر شمارش شود، ساختار زیر در پیوست ۳ در نظر گرفته می شود:

احتمال	تعداد ۰ : ۱۲۳۴
$pq^3 + 2p^2q^2 + p^3q$	۰۱۲۱۰ (مجموع : ۰/۰۹)
$q^4 + p^2q^2 + 2p^3q$	۱۰۱۲۰ (مجموع : ۰/۱۵۴)
$2pq^3 + p^2q^2 + p^4$	۰۲۱۰۱ (مجموع : ۰/۶۶۶)
$q^4 + pq^3 + p^3q + p^4$	۱۱۰۱۱ (مجموع : ۰/۷۳)

۰۱۲۱۰ در حالت $p > q$ بر تمام موقعیت های دیگر غالب است؛ زیرا احتمال ۰۱۲۱۰ کمتر از سایر احتمالات دیگر است.

فرض کنید $\Psi(N, p)$ ماکزیمم احتمال حدس درست در بازی باشد.

$$\Psi(4, p) = 1 - (pq^3 + 2p^2q^2 + p^3q) = 1 - [pq^2(q+p) + p^2q(q+p)] = 1 - pq(q+p) = 1 - pq = 1 - p + p^2$$

نتیجه ۱.۵.۳. $\Psi(4, p) = \Psi(3, p)$.

تمام ۲۴ مجموعه مناسب با احتمال ۰۱۲۱۰، ماتریس تصمیم مولد را تولید می کند. یک ماتریس مولد در پیوست ۳ نشان داده شده است. اولین نتیجه از این مولد ارائه می شود:

	۱	۳	۱۲	۱۴
۰	-۱	۱	۱	۰
۰	-۱	۱	۱	۰
۰	۳	۰	۳	۰
۱	۱	۰	-۱	-۱

روش توسعه یافته که در بخش ۱.۳.۳ مورد استفاده قرار می گیرد، ماتریس تصمیم گیری افزوده را ارائه می دهد.

نکته: هر مجموعه مناسب با $das > 4$ راه حل های بهتری را ارائه نمی دهد.

این موضوع را می توان با اجرای برنامه در پیوست آ، با $das = 5, 6, \dots, 16$ انجام داد.

به نظر می‌رسد این‌طور نیست؛ $das = ۵$ مولد ۵۶۰ مجموعه مناسب می‌باشد که ۸۰ مجموعه جدید هستند. هیچ بسط ساده‌ای با یک عنصر از راه‌حل با $das = ۴$ وجود ندارد؛ اما به‌دنبال راه‌حل‌های مینیمال هستیم و آن‌ها ظاهراً به‌صورت منظم رفتار می‌کنند:

تعداد $\circ$	$\circ$	۱	۲	۳	۴
عبارت مرتبط	q^4	$q^3 p$	$q^2 p^2$	qp^3	p^4
$das = ۴$	$\circ$	۱	۲	۱	$\circ$
$das = ۵$	۱	۱	۲	۱	$\circ$
$das = ۶$	۱	۲	۲	۱	$\circ$
$das = ۷$	۱	۳	۲	۱	$\circ$
$das = ۸$	۱	۴	۲	۱	$\circ$
$das = ۹$	۱	۴	۳	۱	$\circ$
$das = ۱۰$	۱	۴	۴	۱	$\circ$
$das = ۱۱$	۱	۴	۵	۱	$\circ$
$das = ۱۲$	۱	۴	۶	۱	$\circ$
$das = ۱۳$	۱	۴	۶	۲	$\circ$
$das = ۱۴$	۱	۴	۶	۳	$\circ$
$das = ۱۵$	۱	۴	۶	۴	$\circ$
$das = ۱۶$	۱	۴	۶	۴	۱

رفتار راه‌حل‌های مینیمال برای $N = ۵$ خیلی ساده نیست، همان‌طور که انتظار می‌رود. (در بخش بعدی مشاهده خواهید کرد).

۶.۳ بازی کلاه پنج‌نفره

۱.۶.۳ بازی کلاه پنج‌نفره متقارن

مولد AS در پیوست آ اجرا می‌شود. ابتدا $N = ۵$ ، $p = ۰/۵$ ، ۶ و $۴,۵$ das را در نظر بگیرید، مجموعه مناسبی پیدا نمی‌شود. سپس $das = ۷$ را امتحان می‌شود و به‌عنوان نتیجه، یک لیست از ۳۲۰ مجموعه مناسب، هر کدام با احتمال $۰/۲۱۸۷۵$ دریافت می‌شود. ۶ مجموعه مناسب ارائه می‌شود:

i_1	i_2	i_3	i_4	i_5	i_6	i_7	مجموع
$\circ$	۱	۲	۱۵	۲۳	۲۷	۲۸	$۰/۲۱۸۷۵$
$\circ$	۱	۳	۱۴	۲۲	۲۶	۲۹	$۰/۲۱۸۷۵$
$\circ$	۱	۴	۱۵	۲۳	۲۶	۲۹	$۰/۲۱۸۷۵$
$\circ$	۱	۵	۱۴	۲۲	۲۷	۲۸	$۰/۲۱۸۷۵$
$\circ$	۱	۸	۱۵	۲۲	۲۷	۲۹	$۰/۲۱۸۷۵$

احتمال بهینه برابر $25/32 = 0.78125 = 0.78175 - 1$ می باشد. مجموعه مناسب با das مینیمال نیز مجموعه بهینه است.

مجموع احتمالات در مجموعه مناسب ($das = 7, 8, \dots, 32$) $\frac{das}{3^N}$ است. در حالت $N = 5$ با پیروی از روش بخش ۲.۳ برای هر 32^0 مجموعه مناسب، ماتریس تصمیم پیدا می شود.

این روش در پیوست ب، به طور خودکار عمل می کند؛ ماتریس مولد. ماتریس مولد، اولین عنصر را ارائه می دهد:

	۰	۱	۲	۱۵	۲۳	۲۷	۲۸									
-۱	-۱	-۱	۰	۰	۰	۰	۱	۰	۰	۰	۱	۱	۰	۰	-۱	
-۱	-۱	-۱	۰	۰	۰	۰	۱	۰	۰	۰	۱	۱	۰	۰	-۱	
-۱	-۱	-۱	۰	۰	۰	۰	۱	۰	۰	۰	۱	۱	۰	۰	-۱	
۳	-۱	۰	۰	۰	۰	۰	۱	۰	۰	۰	۱	۰	۱	-۱	۰	
۳	-۱	۰	۰	۰	۰	۰	۱	۰	۰	۰	۱	۰	۱	-۱	۰	

ملاحظه ۱.۶.۳. بازیکنان ۱، ۲ و ۳ دارای استراتژی های یکسان می باشند، استراتژی بازیکنان ۴ و ۵ نیز یکسان می باشد.

روش توسعه یافته مانند بخش ۱.۳.۳، ماتریس تصمیم افزوده را می دهد.

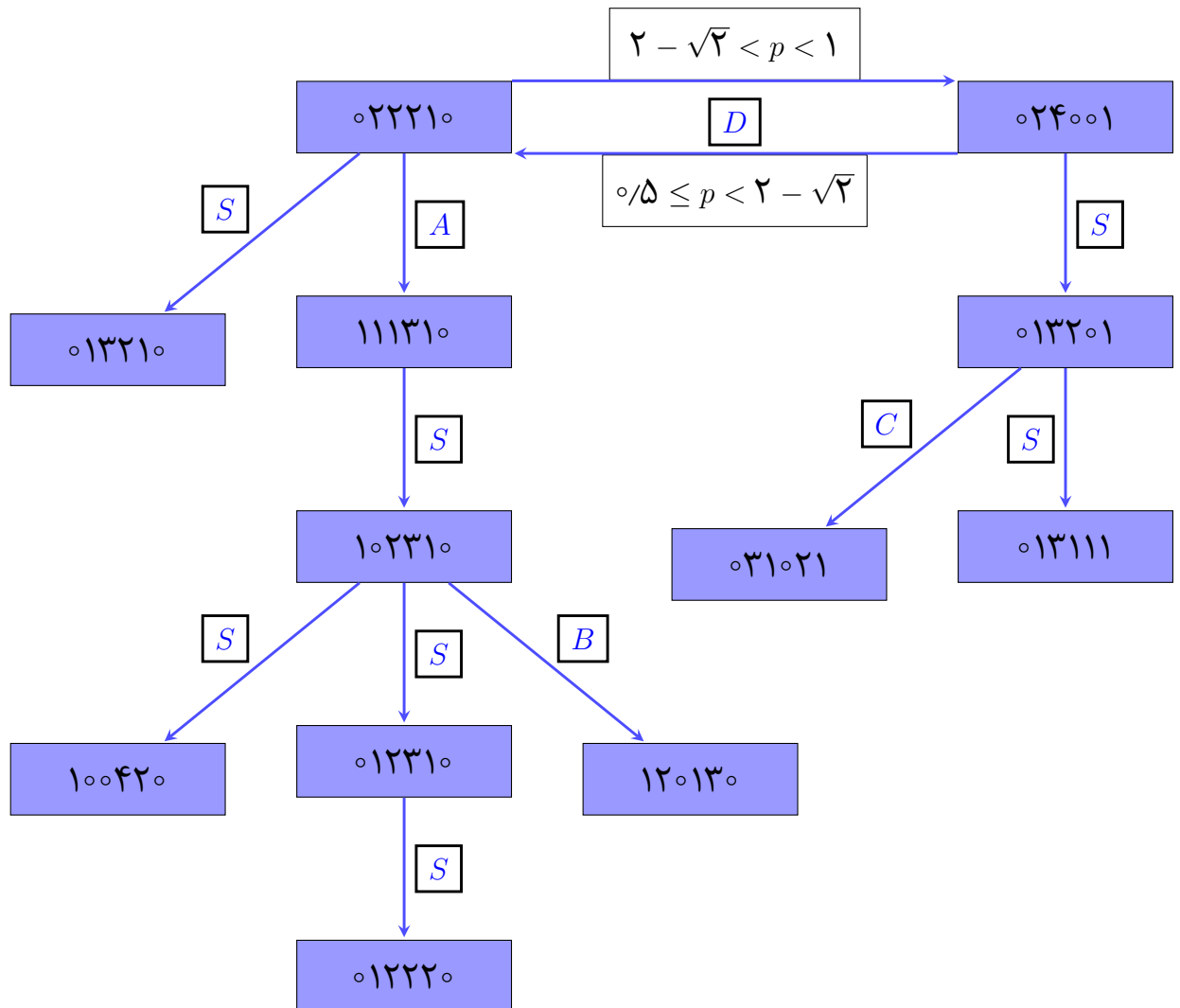
۲.۶.۳ بازی کلاه پنج نفره نامتقارن

برنامه VBA-Excel از پیوست آ، با پارامترهای $N = 5$ و $p = 0.9$ اجرا می شود. تنظیم $das = 7$ ، 32^0 مجموعه مناسب را تولید می کند (برای لیستی مرتب شده و انتخاب شده، پیوست ج را مشاهده کنید). مجموعه مناسب و تعداد صفرها در هر عنصر در مجموعه مناسب، مستقل از p می باشد. ساختار زیر را در پیوست ج، در نظر بگیرید:

احتمال	(تعداد 0) 012345
$2pq^4 + 2p^2q^3 + 2p^3q^2 + p^4q$	022210
$pq^4 + 3p^2q^3 + 2p^3q^2 + p^4q$	013210
$q^5 + pq^4 + p^2q^3 + 3p^3q^2 + p^4q$	111310
$q^5 + 2p^2q^3 + 3p^3q^2 + p^4q$	102310
$pq^4 + 2p^2q^3 + 3p^3q^2 + p^4q$	012310
$2pq^4 + 2p^2q^3 + 2p^3q^2 + 2p^4q$	012220
$q^5 + 4p^3q^2 + 2p^4q$	100420
$q^5 + 2pq^4 + p^2q^3 + 3p^4q$	120130
$2pq^4 + 4p^2q^3 + 3p^5$	024001
$pq^4 + 3p^2q^3 + 2p^3q^2 + p^5$	013201
$pq^4 + 3p^2q^3 + p^3q^2 + p^4q + p^5$	013111
$3pq^4 + p^2q^3 + 2p^4q + p^5$	031021

نمودار زیر نشان‌دهنده روابط غالب بین این ۱۲ کلاس مختلف است.

شکل ۱.۳: روابط بین ۱۲ کلاس



یک کلاس بر کلاس دیگری غالب است اگر احتمال آن کلاس، مستقل از ارزش p ، کمتر از احتمال کلاس دیگر باشد.

نمادگذاری:

- یک فلش با نماد S به این معنی است که بیت‌ها یک یا چند شیفت به سمت چپ داشته‌اند (از یک کلاس غالب)، در نتیجه الگوی یک کلاس غالب می‌باشد (کلاس‌هایی که به سمت چپ شیفت دارند دارای ارزش کمتری هستند؛ زیرا $p > q$).
- از «>» به عنوان نماد غالب (چیرگی) استفاده می‌شود.

فلش‌های A, B, C و D نیاز به توجه دارند.

$$A: 111310 > 022210$$

قضیه ۱.۶.۳. اگر $p > q > 0$ باشد؛ آن‌گاه داریم:

$$-q^5 + pq^4 + p^2q^3 - p^3q^2 < 0.$$

برهان.

$$-q^5 + pq^4 + p^2q^3 - p^3q^2 = -q^2(p-q)^2 < 0.$$

□

$$B \text{ و } C: 102310 > 120310 \text{ و } 013201 > 031021$$

قضیه ۲.۶.۳. اگر $p > q > 0$ باشد؛ آن‌گاه داریم:

$$-2pq^4 + 2p^2q^3 + 2p^3q^2 - 2p^4q < 0.$$

برهان.

$$-2pq^4 + 2p^2q^3 + 2p^3q^2 - 2p^4q = -2pq(p-q)^3 < 0.$$

□

D : باید ثابت کنیم که:

قضیه ۳.۶.۳. اگر $1 < p < 2 - \sqrt{2}$ باشد؛ آن‌گاه $024001 > 022210$ و اگر $2 - \sqrt{2} < p < 5/8$ باشد؛ آن‌گاه $022210 > 024001$ می‌باشد.

برهان.

$$\begin{aligned} -2p^2q^3 + 2p^3q^2 + p^4q - p^5 &= p^2(2p^3 - 9p^2 + 8p - 2) = p^2(2p-1)(p^2 - 4p + 2) \\ &= p^2(2p-1)(p-2-\sqrt{2})(p-2+\sqrt{2}). \end{aligned}$$

□

فرض کنید $\Psi(N, p)$ ماکزیمم احتمال حدس درست در بازی کلاه مدنظر باشد.

اگر $1 < p < 2 - \sqrt{2}$ باشد؛ آن‌گاه $024001 > 022210$ است و

$$\Psi(5, p) = 1 - (2pq^4 + 2p^2q^3 + 2p^3q^2 + p^4q) = 1 - 2p + 6p^2 - 8p^3 + 5p^4 - p^5.$$

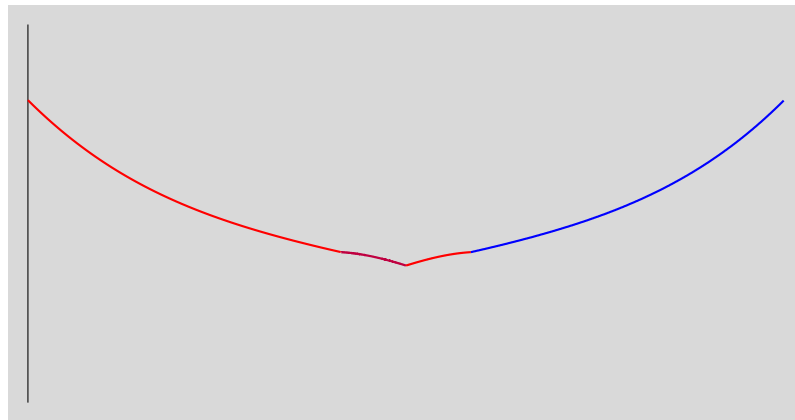
اگر $5/8 < p < 2 - \sqrt{2}$ باشد؛ آن‌گاه $022210 > 024001$ است و

$$\Psi(5, p) = 1 - (2pq^4 + 4p^2q^3 + p^5) = 1 - 2p + 4p^2 - 4p^4 + p^5.$$

با تغییر نقش p و q داریم:

قضیه ۴.۶.۳.

$$\Psi(5, p) = \begin{cases} 1 - p + 2p^2 - 2p^3 + p^5 & (0 \leq p \leq \sqrt{2} - 1) \\ 5p - 10p^2 + 6p^3 + p^4 - p^5 & (\sqrt{2} - 1 \leq p \leq 0.5) \\ 1 - 2p + 4p^2 - 4p^4 + p^5 & (0.5 \leq p \leq 2 - \sqrt{2}) \\ 1 - 2p + 6p^2 - 8p^3 + 5p^4 + p^5 & (2 - \sqrt{2} \leq p \leq 1) \end{cases}$$



شکل ۲.۳: گراف $\Psi(5, p)$

ملاحظه ۲.۶.۳. نقطهٔ مینیمم است و $\Psi(5, p)$ بر حسب 0.5 ، $\sqrt{2} - 1$ و $2 - \sqrt{2}$ قابل تشخیص نیست.

حالت $N = 5$ دارای 32° مجموعهٔ مناسب می‌باشد.

با استفاده از پیوست آ داریم:

وقتی $0.5 < p < 2 - \sqrt{2}$ قرار دارد: 1° مجموعهٔ مناسب بهینه داریم،

وقتی $1 < p < 2 - \sqrt{2}$ قرار دارد: 3° مجموعهٔ مناسب بهینه داریم،

وقتی $p = 2 - \sqrt{2}$ است: 4° مجموعهٔ مناسب بهینه داریم (اجتماع دو مجموعهٔ مناسب قبل).

ماتریس مولد در پیوست ب را استفاده می‌کنیم، اولین عنصر در هر سه حالت اتفاق می‌افتد:

حالت $0.5 < p < 2 - \sqrt{2}$:

	۰	۷	۱۱	۱۹	۲۸	۲۹	۳۰								
-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
-۱	۰	۰	۱	۰	۱	۰	۰	۰	۱	۰	۰	۰	۰	۳	-۱
-۱	۰	۰	۱	۰	۱	۰	۰	۰	۱	۰	۰	۰	۰	۳	-۱

ملاحظه ۳.۶.۳. استراتژی بازیکنان ۱، ۲ و ۳ یکسان می‌باشد و بازیکنان ۴ و ۵ نیز استراتژی یکسان دارند.

حالت $1 < p < 2 - \sqrt{2}$:

	۱	۶	۱۴	۲۲	۲۴	۲۷	۲۹								
۰	-۱	۰	۰	۰	۳	۰	۱	۰	۰	۱	۰	۱	-۱	۰	
۰	-۱	۰	۰	۰	۳	۰	۱	۰	۰	۱	۰	۱	-۱	۰	
۰	-۱	۱	۰	۰	۱	۰	۰	۰	۱	۰	-۱	۱	۰	-۱	
۰	-۱	۱	۰	۰	۱	۰	۰	۰	۱	۰	-۱	۱	۰	-۱	
۱	۰	۰	-۱	۰	۰	-۱	۰	۰	۰	-۱	-۱	۱	۱	۰	

ملاحظه ۴.۶.۳. بازیکنان ۱ و ۲ استراتژی یکسان دارند، استراتژی بازیکنان ۳ و ۴ نیز یکسان می‌باشد و بازیکن ۵ نیز استراتژی خودش را دارد.

تعداد مجموعه‌های مناسب در حالت $p = 2 - \sqrt{2}$ ، از اجتماع 10 مجموعه بهینه در حالت $2 - \sqrt{2} < p < 5/8$ و 30 مجموعه بهینه در حالت $1 < p < 2 - \sqrt{2}$ به دست می‌آید.

نکته: هر مجموعه مناسب با $das > 7$ راه‌حل‌های بهتر را ارائه نمی‌دهد. این موضوع را می‌توان با اجرای برنامه در پیوست آ، با $das = 8, 9, \dots, 32$ انجام داد.

وقتی $1 < p < 2 - \sqrt{2}$ باشد، نتایج را به‌طور منظم در $N = 4$ و 3 در بر می‌گیریم. اما وقتی $2 - \sqrt{2} < p < 5/8$ قرار گرفته باشد، با $das = 7$ و الگوی 10100240 شروع می‌کنیم و با $das = 31$ الگوی $0 - 5 - 10 - 10 - 5 - 1$ را دریافت می‌کنیم. در مکانی (در یک جایی) باید p^5 را حذف کنیم. پاسخ به این سوال را می‌توان در جدول ۲.۳ پیدا کرد.

توضیحات مربوط به جدول ۲.۳:

گام ۱۷ تا ۱۸ جدول مورد نظر مثبت است:

$$2q^2p^3 - p^5 = p^3(2q^2 - p^2) = p^3(p^2 - 4p + 2) = p^3(p - 2 - \sqrt{2})(p - 2 + \sqrt{2}) > 0.$$

همچنین $q^3p^2 < 2q^2p^3 - p^5 < q^2p^3$ می‌باشد؛ زیرا $2q^2p^3 - p^5$ در جدول ۲.۳ بین عبارات q^3p^2 و q^2p^3 قرار دارد.

جدول ۲.۳

تعداد $\circ$	$\circ$	۱	۲	۳	۴	۵
عبارت مرتبط	q^5	$q^4 p$	$q^3 p^2$	$q^2 p^3$	qp^4	p^5
$das = 7$	$\circ$	۲	۴	$\circ$	$\circ$	۱
$das = 8$	۱	۲	۴	$\circ$	$\circ$	۱
$das = 9$		۳				
$das = 10$		۴				
$das = 11$	۱	۵	۴	$\circ$	$\circ$	۱
$das = 12$			۵			
$das = 13$			۶			
$das = 14$			۷			
$das = 15$			۸			
$das = 16$			۹			
$das = 17$	۱	۵	۱۰	$\circ$	$\circ$	۱
$das = 18$	۱	۵	۱۰	۲	$\circ$	$\circ$
$das = 19$				۳		
$das = 20$				۴		
$das = 21$				۵		
$das = 22$				۶		
$das = 23$				۷		
$das = 24$				۸		
$das = 25$				۹		
$das = 26$	۱	۵	۱۰	۱۰	$\circ$	$\circ$
$das = 27$					۱	
$das = 28$					۲	
$das = 29$					۳	
$das = 30$					۴	
$das = 31$	۱	۵	۱۰	۱۰	۵	$\circ$
$das = 32$	۱	۵	۱۰	۱۰	۵	$\circ$

۷.۳ مجموعه‌های پوششی

در این فصل از مجموعه‌های پوششی به صورت قابل توجهی استفاده شده است. بحث بعدی در رابطه با حالت متقارن بازی کلاه می‌باشد. در بخش‌های قبل، مجموعه‌های مناسب در S ، مجموعه S - کد، جایی که کدهای پوششی از شعاع $N - 1$ هستند را مورد بررسی قرار دادیم.

ملاحظه ۱.۷.۳. در حالت $p = \frac{1}{q}$ ، $p = \frac{das}{q^N}$ (مجموعه مناسب) است. همچنین ماکزیمم

احتمال در بازی مدنظر $1 - \frac{das^*}{2^N}$ می باشد، که das^* نشان دهنده کوچک ترین das است. با توجه به نتایج [۱۷] و [۲۲] ماکزیمم احتمال در بازی کلاه نیز برابر $1 - \frac{K(N,1)}{2^N}$ می باشد؛ که $K(N,1)$ کوچک ترین اندازه یک کد پوششی دودویی از شعاع ۱ است. بنابراین در حالت متقارن، ابعاد مینیمال از یک مجموعه مناسب با کدهای پوششی از شعاع $N-1$ برابر $K(N,1)$ است که $K(N,1)$ نشان دهنده کوچک ترین کد پوششی دودویی از شعاع ۱ است.

۸.۳ پیچیدگی محاسباتی

در این بخش، تعداد استراتژی های لازم برای حل مسئله کلاه با N بازیکن و دو رنگ را بررسی می کنیم. هریک از N بازیکن دارای 2^{N-1} شرایط ممکن برای مشاهده است و در هر وضعیت سه حدس احتمالی «پاس»، «سفید» یا «سیاه» وجود دارد؛ بنابراین $(3^{2^{N-1}})^N$ حالت ممکن (استراتژی) داریم.

کرزیوکوسکی [۲۰] نشان داد که $(3^{2^{N-1}-2})^N$ استراتژی برای بررسی کافی است. در روش مجموعه مناسب باید هر $\{i_1, i_2, \dots, i_{das}\}$ که $0 \leq i_1 < i_2 < \dots < i_{das} \leq 2^N - 1$ است، مورد بررسی قرار گیرد. بنابراین تعداد استراتژی ها برابر با تعداد زیرمجموعه های با اندازه das از $\{0, 1, \dots, 2^N - 1\}$ می باشد.

$$\text{تعداد استراتژی ها} = \binom{2^N}{das}$$

برای به دست آوردن قدرت روش مجموعه مناسب، در جدول زیر تعدادی از استراتژی ها را (در سه روش سعی و خطا، کرزیوکوسکی و مجموعه مناسب) مقایسه می کنیم.

$N :$	$das :$	$(3^{2^{N-1}})^N$	$(3^{2^{N-1}-2})^N$	$\binom{2^N}{das}$
۳	۲	۵۳۱۴۴۱	۷۲۹	۲۸
۴	۴	$1/8E + 15$	$2/8E + 11$	۱۸۲۰
۵	۷	$1/5E + 38$	$2/5E + 33$	۳۳۶۵۸۵۶
۶	۱۲	$4/0E + 91$	$7/6E + 85$	$3/2E + 12$
۷	۱۶	$5/6E + 213$	$1/2E + 207$	$9/3E + 19$
۸	۳۲	$3/7E + 488$	$8/7E + 480$	$5/8E + 40$
۹	۶۲	$1/9E + 1099$	$5/0E + 1090$	$6/4E + 80$

توجه: محاسبات ($N \leq 5$) بر روی یک لپ تاپ استاندارد انجام شده است، وقتی $N \geq 6$ است داده das در [۱۷] یافت می شود.

۹.۳ بررسی اجمالی

جدول‌های بعدی مروری بر برخی از نتایج به‌دست‌آمده می‌باشد:

N	$\Psi(N, p)$	das	$nasopt$
۲	$q \quad (0 < p \leq 0/5)$ $p \quad (0/5 \leq p < 1)$	۲	$۶ \quad (p = 0/5)$ $۲ \quad (p \neq 0/5)$
۳	$1 - pq \quad (0 < p < 1)$	۲	$۴ \quad (p = 0/5)$ $۳ \quad (p \neq 0/5)$
۴	$1 - pq \quad (0 < p < 1)$	۲	$۴۰ \quad (p = 0/5)$ $۲۴ \quad (p \neq 0/5)$
۵	$\begin{cases} 1 - p + 2p^2 - 2p^3 + p^5 & (0 \leq p \leq \sqrt{2} - 1) \\ 5p - 10p^2 + 6p^3 + p^4 - p^5 & (\sqrt{2} - 1 \leq p \leq 0/5) \\ 1 - 2p + 4p^2 - 4p^3 + p^4 & (0/5 \leq p \leq 2 - \sqrt{2}) \\ 1 - 2p + 6p^2 - 8p^3 + 5p^4 - p^5 & (2 - \sqrt{2} \leq p \leq 1) \end{cases}$	۷	$۳۲۰ \quad (p = 0/5)$ $۱۰ \quad (\sqrt{2} - 1 < p < 2 - \sqrt{2}, p \neq 0/5)$ $۳۰ \quad (0 < p < \sqrt{2} - 1 \vee 2 - \sqrt{2} < p < 1) \wedge (p \neq 0/5)$ $۴۰ \quad (p < \sqrt{2} - 1 \vee p = 2 - \sqrt{2})$

N	$N = ۲, ۳, ۴ : 0/5 \leq p \leq 1; 5A : 0/5 \leq p \leq 2 - \sqrt{2}; 5B : 2 - \sqrt{2} \leq p \leq 1$															
۲	۰	*														
	۱	۱														
۳	۰	-۱	۱	۰												
	۰	-۱	۱	۰												
	۱	۰	۰	-۱												
۴	۰	-۱	۰	-۱	۱	۰	۱	۰								
	۰	-۱	۰	-۱	۱	۰	۱	۰								
	۰	*	۰	۰	۰	۰	*	۰								
	۱	۱	۰	۰	۰	۰	-۱	-۱								
۵	-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
A	-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
	-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
	-۱	۰	۰	۱	۰	۰	۰	-۱	۰	۰	۰	-۱	۱	۱	۱	۰
	-۱	۰	۰	۱	۰	۱	۰	۰	۰	۱	۰	۰	۰	۰	*	-۱
	-۱	۰	۰	۱	۰	۱	۰	۰	۰	۱	۰	۰	۰	۰	*	-۱
۵	۰	-۱	۰	۰	۰	۰	*	۰	۱	۰	۰	۱	۰	۱	-۱	۰
B	۰	-۱	۰	۰	۰	۰	*	۰	۱	۰	۰	۱	۰	۱	-۱	۰
	۰	-۱	۱	۰	۰	۰	۱	۰	۰	۰	۱	۰	-۱	۱	۰	-۱
	۰	-۱	۱	۰	۰	۰	۱	۰	۰	۰	۱	۰	-۱	۱	۰	-۱
	۱	۰	۰	-۱	۰	۰	۰	-۱	۰	۰	۰	-۱	-۱	۱	۱	۰
	۱	۰	۰	-۱	۰	۰	۰	-۱	۰	۰	۰	-۱	-۱	۱	۱	۰

فصل ۴

نتیجه‌گیری و پیشنهاد

۱.۴ نتیجه‌گیری

در این پایان‌نامه دو کران بالا روی ابعاد ماکزیمم d_n برای بازی‌های ساده با n بازیکن معرفی شده است. کران‌ها مربوط به کدهای دودویی بوده و این کران‌ها باعث بهبود در کران بالای $|L^M|$ معرفی شده توسط تیلور و زویکر [۲۹] می‌شوند.

اخیراً کران پایین $d_n \geq \frac{1}{n} \binom{n}{\lfloor \frac{n}{4} \rfloor}$ به $d_n \geq 2^{\frac{n}{4}-1}$ بهبود یافته است [۱۷]. همچنین کران بالای $d_n \leq \binom{n}{\lfloor \frac{n}{4} \rfloor} = (1 - o(1)) \sqrt{\frac{2}{\pi n}} 2^n$ که توسط تیلور و زویکر ثابت شده است به کران بالای $d_n \leq \frac{2^n}{n+1}$ در حالت $n = 2^m - 1$ بهبود بخشیده شده است. این بهبود اخیر [۲۴] برای کران پایین d_n به صورت زیر می‌تواند بیان شود:

$$2^{\frac{n}{4}-1} \rightarrow \frac{1}{n} \binom{n}{\lfloor \frac{n}{4} \rfloor} = (1 - o(1)) \sqrt{\frac{2}{\pi n}} \frac{2^n}{n} \leq d_n. \quad (1.4)$$

از طرف دیگر، یکی از کران‌های بالا در این فصل، بهبود زیر را برای کران بالا در حالت $n = 2^m - 1$ معرفی می‌کند:

$$d_n \leq \frac{2^n}{n+1} \leftarrow (1 - o(1)) \sqrt{\frac{2}{\pi n}} 2^n = \binom{n}{\lfloor \frac{n}{4} \rfloor}. \quad (2.4)$$

توجه کنید که مقایسه کران‌های بالا و پایین جدید، فاصله ممکن برای بُعد بازی‌های ساده به صورت قابل ملاحظه‌ای نسبت به قبل کمتر شده است.

همان‌طور که قبلاً ذکر شد، برای دستیابی به یک کران بالا با ابعاد توضیح داده شده در قضیه ۱.۴.۲، به یک کد دودویی با شعاع ۱ نیاز است تا بتوان L^M را پوشش داد.

سؤال ۱.۱.۴. این می‌تواند یک سؤال جالب باشد که آیا کران بالای ارائه شده در قضیه ۱.۴.۲ با توجه به ساختار خاص L^M می‌تواند بهبود داده شود؟

نکته اصلی در کران‌های بالای ارائه‌شده، تجزیه L^M به صورت اجتماع مجموعه ائتلاف‌های ماکسیمال باخت است که هر کدام می‌تواند به عنوان یک بازی وزنی تعریف شود. این کار می‌تواند به راه‌های مختلفی انجام شود و کاملاً محتمل است که تجزیه‌های قوی‌تر و هدفمندتر از آنچه در این پایان‌نامه ارائه شد، وجود داشته باشد. این یک موضوع باز برای پیدا کردن تجزیه‌های بهتر است.

۲.۴ پیشنهاد

همان‌طور که مشاهده کردید برای به دست آوردن مجموعه مناسب ابتدا بایستی das بهینه را به دست آورد. با استفاده از روشی که ارائه شده است اگر تعداد بازیکنان زیاد شود نیاز به زمان و محاسبات بیشتری داریم؛ لذا برای کم کردن مراحل الگوریتم و صرفه‌جویی در زمان، پیشنهاد می‌شود برای محاسبه das بهینه، اندازه کد پوششی را به دست آورده و از آن استفاده شود. برای اندازه کد پوششی بهتر است از کران‌های به دست آمده برای K_n در نتیجه ۱.۳.۲ استفاده شود. K_n نشان‌دهنده کوچک‌ترین اندازه از یک کد پوششی دودویی به طول n با شعاع پوششی ۱ می‌باشد.

پیوست آ

مولد AS

Alt F11: تغییر وضعیت بین Excel و VBA؛

F5: اجرای برنامه (در VBA).

برنامه، مقادیر پارامترهای N ، P و das را لازم دارد و مجموعه‌های مناسب (AS) را محاسبه می‌کند.

جدول زیر، مقادیر صحیح برای $N = 2, 3, 4$ و ۵ را نشان می‌دهد:

N	۲	۳	۴	۵
das	۲	۳	۴	۷

وفق دادن برنامه وقتی N تغییر می‌کند: $0 \leq i_1 < i_2 < \dots < i_{das} \leq H$ نیاز داریم.

```
Sub adequate_sets()
```

```
'' this program: shows adequate sets (AS)
```

```
'' calculates sum of probabilities in AS
```

```
'' counts number of zero's in each binary element
```

```
'' parameters required: N, p, das
```

```
Dim d() As Integer
```

```
Dim m() As Integer
```

```
Dim i() As Integer
```

```
Dim y() As Integer
```

```
Dim check() As Integer
```

```
p = 0.9
q = 1 - p
N = 4
H = 2 ^ N - 1
das = 4

ReDim d(1 To N) As Integer
ReDim m(0 To N - 1, 0 To H) As Integer
ReDim i(1 To das) As Integer
ReDim y(0 To N - 1, 0 To H) As Integer
ReDim check(0 To H) As Integer

" for each number from 0 to H:
" first calculate binary digits d(.)
" and then the scodes
" fill matrix m with scodes

For num = 0 To H
g = num
For z = 1 To N
d(z) = g Mod 2
g = (g - d(z)) / 2
Next z
For k = 1 To N
scode = 0
For t = 1 To k - 1
scode = scode + d(t) * (2 ^ (t - 1))
Next t
For u = k + 1 To N
scode = scode + d(u) * (2 ^ (u - 2))
Next u
m(k - 1, num) = scode
Next k
Next num
```

```

x=0 "x: row in Excel where result is displayed
'adequate set: {i_1, i_2,\dots,i_{das}\}
For i1 = 0 To H - 3
i(1) = i1
"VBA-EXCEL can't handle with an array in for to next
For i2 = i1 + 1 To H - 2
i(2) = i2
For i3 = i2 + 1 To H - 1
i(3) = i3
For i4 = i3 + 1 To H
i(4) = i4

"check on adequate set property:
For k = 0 To H
check(k) = 0
f = 1
For s = 0 To N - 1
For j = 1 To das
If m(s, i(j)) = m(s, k) Then y(s, i(j)) = 0 Else y(s, i(j)) = 1
f = f * y(s, i(j))
Next j
Next s
If f = 0 Then check(k) = 1
Next k

state = 1

For k = 0 To H
state = state * check(k)
Next k

" state=1 means: we found an adequate set
If state = 1 Then x = x + 1

" next row in Excel; show elements of adequate set: Cells(row,column).

```

```

For k = 1 To das
If state = 1 Then Cells(x, k) = i(k)
Next k
" Sum: sum of probabilities in AS
Sum = 0
If state = 1 Then
For k = 1 To das
g = i(k)
For z = 1 To N
d(z) = g Mod 2
g = (g - d(z)) / 2
Next z
count = 0 " count of zero's in element of adequate set
For t = 1 To N
If d(t) = 0 Then count = count + 1
Next t
term = p ^ count * q ^ (N - count)
Sum = Sum + term
Cells(x, 1 + das) = Sum
Cells(x, das + k + 1) = count
Next k
End If

Next i4
Next i3
Next i2
Next i1
End Sub

```

پیوست ب

ماتریس تصمیم مولد.

تفسیر EXCEL/VBA:

Alt F11: تغییر وضعیت بین Excel و VBA؛

F5: اجرای برنامه (در VBA).

برنامه، مقادیر پارامترهای N ، P ، das (بُعد عناصر مجموعه مناسب)، و $nasopt$ (تعداد عناصر در مجموعه مناسب بهینه) را لازم دارد.

مولد برای هر مجموعه مناسب بهینه در حداقل یک ماتریس تصمیم محاسبه می کند. جدول زیر، مقادیر صحیح را برای $N = 2, 3, 4$ و ۵ نشان می دهد:

مستقل از P		$p = 0/5$	$p > 0/5$
N	das	nasopt = nas	nasopt
۲	۲	۶	۲
۳	۲	۴	۳
۴	۴	۴۰	۲۴
۵	۷	۳۲۰	$1^0 \quad (0/5 < p < 2 - \sqrt{2})$ $3^0 \quad (2 - \sqrt{2} < p < 1)$ $4^0 \quad (p = 2 - \sqrt{2})$

وفق دادن برنامه وقتی N تغییر می کند: $0 \leq i_1 < i_2 < \dots < i_{das} \leq H$ نیاز داریم.

```
Sub Decision_Matrix_Generator()
```

```
Dim c() As Integer
```

```
Dim d() As Integer
```

```
Dim m() As Integer
```

```
Dim m2() As Integer
```

```
Dim i() As Integer
```

```
Dim y() As Integer
```

```

Dim check() As Integer
Dim rdec() As Integer
Dim rbin() As Integer
Dim s2() As Integer

p = 0.9
q = 1 - p
N = 4 H = 2 ^ N - 1
das = 4
nasopt = 24 'number of optimal sets

ReDim c(1 To N) As Integer
ReDim d(1 To N) As Integer
ReDim m(0 To N - 1, 0 To H) As Integer
ReDim m2(1 To nasopt, 1 To N, 0 To 2 ^ (N - 1) - 1) As Integer
ReDim i(1 To das) As Integer
ReDim y(0 To N - 1, 0 To H) As Integer
ReDim check(0 To H) As Integer
ReDim rdec(1 To nasopt, 1 To das) As Integer
ReDim rbin(1 To nasopt, 1 To das, 1 To N) As Integer
ReDim s2(1 To nasopt, 1 To das, 1 To N) As Integer

" calculate reversed bincode d(.), bincode c(.)
" scodes and fill matrix m with scodes
For num = 0 To H
    g = num
    For z = 1 To N
        d(z) = g Mod 2
        g = (g - d(z)) / 2
    Next z
    For z = 1 To N
        c(z) = d(N + 1 - z)
    Next z
    For k = 1 To N

```

```

scode = 0
For t = 1 To k - 1
scode = scode + c(t) * (2 ^ (N - t - 1))
Next t
For u = k + 1 To N
scode = scode + c(u) * (2 ^ (N - u))
Next u
m(k - 1, num) = scode
Next k
Next num
x = 0    "x: row in Excel where result is displayed
"Search of AS:
For i1 = 0 To H - 3
i(1) = i1
For i2 = i1 + 1 To H - 2
i(2) = i2
For i3 = i2 + 1 To H - 1
i(3) = i3
For i4 = i3 + 1 To H
i(4) = i4
"check of AS property:
For k = 0 To H
check(k) = 0
f = 1
For s = 0 To N - 1
For j = 1 To das
If m(s, i(j)) = m(s, k) Then y(s, i(j)) = 0 Else y(s, i(j)) = 1
f = f * y(s, i(j))
Next j
Next s
If f = 0 Then check(k) = 1
Next k

state = 1

```

```

For k = 0 To H
state = state * check(k)
Next k
"state=1: adequate set

Sum = 0
"sum of probabilities in AS

If state = 1 Then "start long if
For k = 1 To das
g = i(k)
For z = 1 To N
d(z) = g Mod 2
g = (g - d(z)) / 2
Next z
For z = 1 To N
c(z) = d(N + 1 - z)
Next z
count = 0 " count of zero's in members of AS
For t = 1 To N
If c(t) = 0 Then count = count + 1
Next t
term = p ^ count * q ^ (N - count)
Sum = Sum + term
Next k
" fill the decimal codes rdec(.,.), with adequate set values
" optimal solution: N=4: sum=pq
If Sum = p * q Then
x = x + 1
For k = 1 To das
rdec(x, k) = i(k)
Next k
End If
" fill the binary codes rbin(.,.) using decimal codes rdec:

```

```
For a = 1 To nasopt
For j = 1 To das
g = rdec(a, j)
For z = 1 To N
d(z) = g Mod 2
g = (g - d(z)) / 2
Next z
For z = 1 To N
rbin(a, j, z) = d(N + 1 - z)
Next z
Next j
Next a
" fill s2(.,.)with scodes:
For a = 1 To nasopt
For j = 1 To das
For k = 1 To N
scode = 0
For t = 1 To k - 1
scode = scode + rbin(a, j, t) * (2 ^ (N - t - 1))
Next t
For u = k + 1 To N
scode = scode + rbin(a, j, u) * (2 ^ (N - u))
Next u
s2(a, j, k) = scode
Next k
Next j
Next a
" fill m2(.,.), the decision matrix:
" start with zero's in the matrix:
For a = 1 To nasopt
For k = 1 To N
For j = 0 To 2 ^ (N - 1) - 1
m2(a, k, j) = 0
Next j
```

```

Next k
Next a
"use bit map  $b \rightarrow 2^b - 1$  (see section 1.1)
"we use code 3 when all three guesses ( -1,0 and 1) are correct
For a = 1 To nasopt
For k = 1 To N
For j = 1 To das
If  $m2(a, k, s2(a, j, k)) = 0$ 
Then  $m2(a, k, s2(a, j, k)) = 2 * rbin(a, j, k) - 1$ 
Else  $m2(a, k, s2(a, j, k)) = 3$ 
Next j
Next k
Next a
"show header and decision matrix m2:
For a = 1 To nasopt
For k = 1 To N
For j = 1 To das
Cells(6 * a, 1) = a
Cells(6 * a, j + 3) = rdec(a, j)
Next j
Next k
Next a
For a = 1 To nasopt
For k = 1 To N
For j = 0 To  $2^{(N - 1)} - 1$ 
Cells(6 * a + k, j + 1) = m2(a, k, j)
Next j
Next k
Next a
End If " end long if
Next i4
Next i3
Next i2
Next i1

```

YY

End Sub

پیوست پ

آزمون سعی و خطا ($N = ۲, p = ۰/۵$)

ساخت تمام ماتریس‌های تصمیم بهینه، بانضمام ماتریس هم‌ریختشان.

```
Sub Brute_Force_2()  
Dim a(1 To 100, 1 To 2, 1 To 2) As Integer  
p = 0.5  
q = 1 - p  
max = 0.5 'min van AS  
For s1 = 0 To 8  
For s2 = 0 To 8  
Sum = 0  
'case 00  
If v(s1) > -1 And v(s2) > -1 And v(s1) + v(s2) > 0  
Then Sum = Sum + p ^ 2  
'case 01  
If w(s1) > -1 And v(s2) < 1 And Not (w(s1) = 0 And v(s2) = 0)  
Then Sum = Sum + p * q  
'case 10  
If v(s1) < 1 And w(s2) > -1 And Not (v(s1) = 0 And w(s2) = 0)  
Then Sum = Sum + p * q  
'case 11  
If w(s1) < 1 And w(s2) < 1 And Not (w(s1) = 0 And w(s2) = 0)  
Then Sum = Sum + q ^ 2  
If Sum >= max  
Then Count = Count + 1: a(Count, 1, 1) = v(s1): a(Count, 1, 2) = w(s1):  
a(Count, 2, 1) = v(s2): a(Count, 2, 2) = w(s2)
```

```
Next s2
```

```
Next s1
```

```
For r = 1 To Count
```

```
Cells(3 * r - 2, 2) = a(r, 1, 1)
```

```
Cells(3 * r - 2, 3) = a(r, 1, 2)
```

```
Cells(3 * r - 1, 2) = a(r, 2, 1)
```

```
Cells(3 * r - 1, 3) = a(r, 2, 2)
```

```
Next r
```

```
End Sub
```

```
Function v(i) " first ternary digit
```

```
Dim w As Integer
```

```
Dim l As Integer
```

```
l = i
```

```
w = l Mod 3 l = l \ 3
```

```
v = l Mod 3
```

```
v = v - 1
```

```
End Function
```

```
Function w(i) " last ternary digit
```

```
w = i Mod 3
```

```
w = w - 1
```

```
End Function
```

خروجی:

-۱	-۱	-۱	-۱	-۱	-۱	-۱	۰	-۱	۰	-۱	۰
۱	۰	۰	۰	۱	۰	-۱	-۱	-۱	۰	-۱	۱
-۱	۱	-۱	۱	-۱	۱	-۱	۱	۰	-۱	۰	-۱
-۱	۰	-۱	۱	۰	۰	۰	۱	۱	-۱	۱	۰
۰	-۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۱
۱	۱	-۱	-۱	-۱	۱	۱	-۱	۱	۱	-۱	۱
۰	۱	۰	۱	۱	-۱	۱	-۱	۱	-۱	۱	-۱
۰	۱	۱	۱	۰	-۱	۰	۰	۱	-۱	۱	۰
۱	۰	۱	۰	۱	۰	۱	۱	۱	۱	۱	۱
-۱	-۱	۰	-۱	۱	-۱	۰	-۱	۰	۰	۰	۱

پیوست ت

آزمون سعی و خطا ($N = 3; p = 0.5$)

ساخت تمام ماتریس‌های تصمیم بهینه:

```
Sub Brute_Force_3()
Dim m, r, s1, s2, s3, teller As Integer
Dim p, q, max, som As Double
Dim a(1 To 4, 1 To 3, 1 To 4) As Integer
p = 0.5
q = 1 - p

max = 1 - p + p * p
teller = 0

For s1 = 0 To 80
For s2 = 0 To 80
For s3 = 0 To 80
som = 0
'case www
If (t(s1)>-1 And t(s2)>-1 And t(s3)>-1 And t(s1)+t(s2)+t(s3)>0) Then
som=som+p^3

'case wwz
If u(s1)>-1 And u(s2)>-1 And t(s3)<1 And Not (u(s1)= 0 And u(s2)=0
And t(s3)=0) Then som=som+p^2*q
```

'case wzw

If $v(s1) > -1$ And $t(s2) < 1$ And $u(s3) > -1$ And Not ($v(s1) = 0$ And $t(s2) = 0$
And $u(s3) = 0$) Then $som = som + p^2 * q$

'case wzz

If $w(s1) > -1$ And $u(s2) < 1$ And $u(s3) < 1$ And Not ($w(s1) = 0$ And $u(s2) = 0$
And $u(s3) = 0$) Then $som = som + q^2 * p$

'case zww

If $t(s1) < 1$ And $v(s2) > -1$ And $v(s3) > -1$ And Not ($t(s1) = 0$ And $v(s2) = 0$
And $v(s3) = 0$) Then $som = som + p^2 * q$

'case zwz

If $u(s1) < 1$ And $w(s2) > -1$ And $v(s3) < 1$ And Not ($u(s1) = 0$ And $w(s2) = 0$
And $v(s3) = 0$) Then $som = som + q^2 * p$

'case zzw

If $v(s1) < 1$ And $v(s2) < 1$ And $w(s3) > -1$ And Not ($v(s1) = 0$ And $v(s2) = 0$
And $w(s3) = 0$) Then $som = som + q^2 * p$

'case zzz

If $w(s1) < 1$ And $w(s2) < 1$ And $w(s3) < 1$ And Not ($w(s1) = 0$ And $w(s2) = 0$
And $w(s3) = 0$) Then $som = som + q^3$

If $som = \max$ Then $teller = teller + 1$: $a(teller, 1, 1) = t(s1)$:
 $a(teller, 1, 2) = u(s1)$: $a(teller, 1, 3) = v(s1)$: $a(teller, 1, 4) = w(s1)$:
 $a(teller, 2, 1) = t(s2)$: $a(teller, 2, 2) = u(s2)$: $a(teller, 2, 3) = v(s2)$:
 $a(teller, 2, 4) = w(s2)$: $a(teller, 3, 1) = t(s3)$: $a(teller, 3, 2) = u(s3)$:
 $a(teller, 3, 3) = v(s3)$: $a(teller, 3, 4) = w(s3)$

Next s3

Next s2

Next s1

```

Cells(1, 3) = max
Cells(1, 1) = teller
For r = 1 To teller
Cells(5 * r + 1, 2) = a(r, 1, 1)
Cells(5 * r + 1, 3) = a(r, 1, 2)
Cells(5 * r + 1, 4) = a(r, 1, 3)
Cells(5 * r + 1, 5) = a(r, 1, 4)
Cells(5 * r + 2, 2) = a(r, 2, 1)
Cells(5 * r + 2, 3) = a(r, 2, 2)
Cells(5 * r + 2, 4) = a(r, 2, 3)
Cells(5 * r + 2, 5) = a(r, 2, 4)
Cells(5 * r + 3, 2) = a(r, 3, 1)
Cells(5 * r + 3, 3) = a(r, 3, 2)
Cells(5 * r + 3, 4) = a(r, 3, 3)
Cells(5 * r + 3, 5) = a(r, 3, 4)
Next r

```

```

End Sub

```

```

Function t(i) '' ternary digit 1
Dim w As Integer
Dim v As Integer
Dim u As Integer
Dim j As Integer
j = i
w = j Mod 3
j = j \ 3
v = j Mod 3
j = j \ 3
u = j Mod 3
t = j \ 3
t = t - 1
End Function

```

Function u(i) '' ternary digit 2

Dim w As Integer

Dim v As Integer

Dim k As Integer

k = i

w = k Mod 3

k = k \ 3

v = k Mod 3

k = k \ 3

u = k Mod 3

u = u - 1

End Function

Function v(i) '' ternary digit 3

Dim w As Integer

Dim l As Integer

l = i

w = l Mod 3

l = l \ 3

v = l Mod 3

v = v - 1

End Function

Function w(i) '' ternary digit 4

w = i Mod 3

w = w - 1

End Function

خروجی:

۱-	○	○	۱
۱-	○	○	۱
۱-	○	○	۱
○	۱-	۱	○
○	۱-	۱	○
۱	○	○	۱-
○	۱	۱-	○
۱	○	○	۱-
○	۱-	۱	○
۱	○	○	۱-
○	۱	۱-	○
○	۱	۱-	○

پیوست ث

$N = 4$ ، لیست مرتب شده، $p = 0/9$ ، $nas = 40$

$nasopt = 24$

i1	i2	i3	i4	sum	0, bin.i1	0, bin.i2	0, bin.i3	0, bin.i4
۱	۳	۱۲	۱۴	۰, ۰۹	۳	۲	۲	۱
۱	۵	۱۰	۱۴	۰, ۰۹	۳	۲	۲	۱
۱	۶	۹	۱۴	۰, ۰۹	۳	۲	۲	۱
۱	۶	۱۰	۱۴	۰, ۰۹	۳	۲	۲	۱
۱	۶	۱۱	۱۲	۰, ۰۹	۳	۲	۱	۲
۱	۷	۱۰	۱۲	۰, ۰۹	۳	۱	۲	۲
۲	۳	۱۲	۱۳	۰, ۰۹	۳	۲	۲	۱
۲	۵	۹	۱۴	۰, ۰۹	۳	۲	۲	۱
۲	۵	۱۰	۱۳	۰, ۰۹	۳	۲	۲	۱
۲	۵	۱۱	۱۲	۰, ۰۹	۳	۲	۱	۲
۲	۶	۹	۱۳	۰, ۰۹	۳	۲	۲	۱
۲	۷	۹	۱۲	۰, ۰۹	۳	۱	۲	۲
۳	۴	۹	۱۴	۰, ۰۹	۲	۳	۲	۱
۳	۴	۱۰	۱۳	۰, ۰۹	۲	۳	۲	۱
۳	۴	۱۱	۱۲	۰, ۰۹	۲	۳	۱	۲

$nasopt = 24$ ، $nas = 4^\circ$ ، $p = 0/9$ ، لیست مرتب شده، $N = 4 \quad 9^\circ$

i1	i2	i3	i4	sum	0, bin.i1	0, bin.i2	0, bin.i3	0, bin.i4
3	5	8	14	0, 09	2	2	3	1
3	6	8	13	0, 09	2	2	3	1
3	7	8	12	0, 09	2	1	3	2
4	5	10	11	0, 09	3	2	2	1
4	6	9	11	0, 09	3	2	2	1
4	7	9	10	0, 09	3	1	2	2
5	6	8	11	0, 09	2	2	3	1
5	7	8	10	0, 09	2	1	3	2
6	7	8	9	0, 09	2	1	3	2
1	2	12	15	0, 154	3	3	2	0
1	4	10	15	0, 154	3	3	2	0
1	6	8	15	0, 154	3	2	3	0
2	4	9	15	0, 154	3	3	2	0
2	5	8	15	0, 154	3	2	3	0
3	4	8	15	0, 154	2	3	3	0
0	3	13	14	0, 666	4	2	1	1
0	5	11	14	0, 666	4	2	1	1
0	6	11	13	0, 666	4	2	1	1
0	7	9	14	0, 666	4	1	2	1
0	7	10	13	0, 666	4	1	2	1
0	7	11	12	0, 666	4	1	1	2
0	1	14	15	0, 73	4	3	1	0
0	2	13	15	0, 73	4	3	1	0
0	4	11	15	0, 73	4	3	1	0
0	7	8	15	0, 73	4	1	3	0

پیوست ج

$N = 5$ ، $p = 0/9$ ، لیست مرتب شده (انتخاب شده).

در جدول زیر، یک لیست از قسمتی از ۳۲۰ مجموعه مناسب انتخاب شده است. مجموع در هر سطر نشان دهنده دو عضو می باشد.

Legend (headers):

$\{i1, \dots, i7\}$: adequate set

Sum: total probability of adequate set

$n1, \dots, n7$: number of zero's in binary representation of $i1, \dots, i7$.

$0, \dots, 5$: number of hits of $0, \dots, 5$ in $\{n1, \dots, n7\}$.

$i1$	$i2$	$i3$	$i4$	$i5$	$i6$	$i7$	sum($p = 0/9$)	$n1$	$n2$	$n3$	$n4$	$n5$	$n6$	$n7$	0	1	2	3	4	5
1	6	15	23	24	26	28	0.08199	4	3	1	1	3	2	2	0	2	2	2	1	0
1	6	11	13	22	23	24	0.08271	4	3	2	2	2	1	3	0	1	3	2	1	0
1	6	11	19	24	28	29	0.08271	4	3	2	2	3	2	1	0	1	3	2	1	0
1	6	10	18	28	29	31	0.08839	4	3	3	3	2	1	0	1	1	1	3	1	0
1	6	12	20	26	27	31	0.08839	4	3	3	3	2	1	0	1	1	1	3	1	0
1	3	5	14	22	24	31	0.08911	4	3	3	2	2	3	0	1	0	2	3	1	0
1	3	9	14	20	26	31	0.08911	4	3	3	2	3	2	0	1	0	2	3	1	0
1	3	7	12	20	26	29	0.08919	4	3	2	3	3	2	1	0	1	2	3	1	0
1	3	11	12	22	24	29	0.08919	4	3	2	3	2	3	1	0	1	2	3	1	0
1	2	7	12	20	27	28	0.14751	4	4	2	3	3	1	2	0	1	2	2	2	0
1	2	11	12	23	24	28	0.14751	4	4	2	3	1	3	2	0	1	2	2	2	0
1	2	3	12	23	24	31	0.16039	4	4	2	3	3	3	0	1	0	0	4	2	0
1	4	5	10	18	24	31	0.16039	4	4	3	3	3	3	0	1	0	0	4	2	0
1	2	4	15	23	24	31	0.20431	4	4	4	1	1	3	0	1	2	0	1	3	0
1	2	8	15	20	27	31	0.20431	4	4	4	1	3	1	0	1	2	0	1	3	0
0	7	11	19	28	29	30	0.59391	5	2	2	2	2	1	1	0	2	4	0	0	1
0	7	13	21	26	27	30	0.59391	5	2	2	2	2	1	1	0	2	4	0	0	1
0	3	5	14	22	25	30	0.60759	5	3	3	2	2	2	1	0	1	3	2	0	1
0	3	6	13	21	26	29	0.60759	5	3	3	2	2	2	1	0	1	3	2	0	1
0	1	3	14	22	26	29	0.66591	5	4	3	2	2	2	1	0	1	3	1	1	1
0	1	5	14	22	27	28	0.66591	5	4	3	2	2	1	2	0	1	3	1	1	1
0	1	2	15	23	27	28	0.72279	5	4	4	1	1	1	2	0	3	1	0	2	1
0	1	4	15	23	26	29	0.72279	5	4	4	1	1	2	1	0	3	1	0	2	1

- [1] Alon, Noga and Spencer, Joel H. *The probabilistic method*. John Wiley & Sons, 1992.
- [2] Berlekamp, Elwyn. *Algebraic coding theory: Revised Edition*. World Scientific, 2015.
- [3] Blum, W. Denksport für hutträger. *Die Zeit, May*, 3:8, 2001.
- [4] Buhler, Joe P. Hat tricks. *The Mathematical Intelligencer*, 24(4):44–49, 2002.
- [5] Burke, Edmund, Gustafson, Steven, and Kendall, Graham. A puzzle to challenge genetic programming. In *European Conference on Genetic Programming*, pages 238–247. Springer, 2002.
- [6] Cheung, Wai-Shun and Ng, Tuen-Wai. A three-dimensional voting system in hong kong. *European Journal of Operational Research*, 236(1):292–297, 2014.
- [7] Cohen, Gérard, Honkala, Iiro, Litsyn, Simon, and Lobstein, Antoine. *Covering codes*, volume 54. Elsevier, 1997.
- [8] Ebert, Todd. Applications of recursive operators to randomness and complexity. 1999.
- [9] Ebert, Todd, Merkle, Wolfgang, and Vollmer, Heribert. On the autoreducibility of random sequences. *SIAM Journal on Computing*, 32(6):1542–1569, 2003.
- [10] Faliszewski, Piotr, Elkind, Edith, and Wooldridge, Michael. Boolean combinations of weighted voting games. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 185–192. International Foundation for Autonomous Agents and Multiagent Systems, 2009.
- [11] Freixas, Josep. The dimension for the european union council under the nice rules. *European Journal of Operational Research*, 156(2):415–419, 2004.

- [12] Freixas, Josep and Marciniak, Dorota. On the notion of dimension and codimension of simple games. *Contributions to Game Theory and Management*, 3(0):67–81, 2010.
- [13] Freixas, Josep and Puente, María Albina. A note about games-composition dimension. *Discrete Applied Mathematics*, 113(2-3):265–273, 2001.
- [14] Gardner, Martin. *The 2nd Scientific American book of mathematical puzzles & diversions*. Simon and Schuster New York, 1961.
- [15] Guo, W, Kasala, S, Rao, M, and Tucker, B. The hat problem and some variations, advances in distribution theory, order statistics, and inference, 459-479. *Stat. Ind. Technol.*, Birkhäuser Boston, 2006.
- [16] Johnson, William B. <http://exoplanet.eu/catalog.php>.
- [17] Kéri, Gerzson. Tables for bounds on covering codes (2011). URL <http://www.sztaki.hu/~keri/codes>.
- [18] Kilgour, D Marc. A formal analysis of the amending formula of canada’s constitution act, 1982. *Canadian Journal of Political Science/Revue canadienne de science politique*, 16(4):771–777, 1983.
- [19] Krzywkowski, Marcin. On the hat problem, its variations, and their applications. *Annales Universitatis Paedagogicae Cracoviensis Studia Mathematica*, 9(1):55–67, 2010.
- [20] Krzywkowski, Marcin. A more colorful hat problem. *Annales Universitatis Paedagogicae Cracoviensis Studia Mathematica*, 10(1):67–77, 2011.
- [21] Kurz, Sascha and Napel, Stefan. Dimension of the lisbon voting rules in the eu council: a challenge and new world record. *Optimization Letters*, 10(6):1245–1256, 2016.
- [22] Lenstra, HW and Seroussi, G. On hats and other covers. Technical report, cs/0509045, 2005.
- [23] Lubell, David. A short proof of sperner’s lemma. *Journal of Combinatorial Theory*, 1(2):299, 1966.

- [24] Olsen, Martin, Kurz, Sascha, and Molinero, Xavier. On the construction of high-dimensional simple games. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence*, pages 880–885. IOS Press, 2016.
- [25] Östergård, Patric RJ and Kaikkonen, Markku K. New upper bounds for binary covering codes. *Discrete Mathematics*, 178(1-3):165–179, 1998.
- [26] Poulos, J. Could you solve this \$1 million hat trick?, abcnews, 2001.
- [27] Robinson, Sara. Why mathematicians now care about their hat color. *The New York Times, Science Times Section*, page D, 5, 2001.
- [28] Taylor, Alan and Zwicker, William. Weighted voting, multicameral representation, and power. *Games and Economic Behavior*, 5(1):170–181, 1993.
- [29] Taylor, Alan D, Zwicker, William S, and Zwicker, William. *Simple games: Desirability relations, trading, pseudoweightings*. Princeton University Press, 1999.

واژه‌نامه انگلیسی به فارسی

A

Additional	افزوده، اضافی
Adequate set	مجموعه مناسب
Asymmetric	نامتقارن

B

Bipartite graph	گراف دوبخشی
Block code	کد بلوکی
Bound	کران
Bounded	کراندار

C

Calculations	محاسبات
Coalition	ائتلاف
Code alphabet	الفبای کد
Code cardinality	اندازه کد
Codeword	کدواژه
Coding theory	نظریه کدگذاری
Communication channel	کانال ارتباطی
Computational complexity	پیچیدگی محاسباتی
Configuration	وضعیت
Corresponding	متناظر

Counterpart	همتا
Covering radius	شعاع پوششی
Cryptology	رمزنگاری

D

Decimal	ده‌دهی
Decision matrix	ماتریس تصمیم
Decoding	کدگشایی
Decompose	تجزیه کردن
Dimension	بُعد
Distinguishable	قابل تشخیص

E

Encoding	کدگذاری
Error correcting code	کد تصحیح‌کننده خطا

F

Flipping	تغییر بیت
----------	-----------

H

Hamming distance	فاصله همینگ
------------------	-------------

I

Improvement	بهبود
Independent	مستقل
Information rate	نرخ اطلاعات

L

Limitation	محدودیت
Linear code	کد خطی
Lower bound	کران پایین

O

Omit	حذف کردن
Optimal	بهینه
Overcome	غلبه کردن

P

Perfect code	کد کامل
--------------	---------

R

Reverse	معکوس
---------	-------

S

Simultaneous	همزمان
--------------	--------

T

Trivial	بدیهی
---------	-------

U

Union	اجتماع
Upper bound	کران بالا

واژه‌نامه فارسی به انگلیسی

ا

Union	اجتماع
Additional	اضافی، افزوده
Code alphabet	الفبای کد
Code cardinality	اندازه کد
Coalition	ائتلاف

ب

Trivial	بدیهی
Dimension	بُعد
Improvement	بهبود
Optimal	بهینه

پ

Computational complexity	پیچدگی محاسباتی
--------------------------	-----------------

ت

Decompose	تجزیه کردن
Flipping	تغییر بیت

ح

Omit حذف کردن

د

Decimal ده‌دهی

ر

Cryptology رمزنگاری

ش

Covering radius شعاع پوششی

غ

Overcome غلبه کردن

ف

Hamming distance فاصله همینگ

ق

Distinguishable قابل تشخیص

ک

Communication channel کانال ارتباطی

Block code کد بلوکی

Error correcting code کد تصحیح‌کننده خطا

Linear code	کد خطی
Perfect code	کد کامل
Encoding	کدگذاری
Decoding	کدگشایی
Codeword	کدواژه
Bound	کران
Upper bound	کران بالا
Lower bound	کران پایین
Bounded	کراندار

گ

Bipartite graph	گراف دوبخشی
-----------------	-------------

م

Decision matrix	ماتریس تصمیم
Corresponding	متناظر
Adequate set	مجموعه مناسب
Calculations	محاسبات
Limitation	محدودیت
Independent	مستقل
Reverse	معکوس

ن

Asymmetric	نامتقارن
Information rate	نرخ اطلاعات
Coding theory	نظریه کدگذاری

و

Configuration وضعیت

ه

Counterpart هم‌تا

Simultaneous هم‌زمان

Abstract

Consider a situation with n agents or players where some of the players form a coalition with a certain collective objective. Simple games are used to model systems that can decide whether coalitions are successful (winning) or not (losing). The dimension of a simple game is the smallest positive integer d such that the simple game can be expressed as the intersection of d threshold functions where each threshold function uses a threshold and n weights. In this thesis, we study the lower and upper bounds for the dimension of a simple game. In this regard, the upper bound by Taylor and Zwicker will be presented which states that the dimension is at most the number of maximal losing coalitions then this bound will be significantly improved to the cardinality of a binary covering code with length n and radius 1.

As another topic studied in this thesis, we explore the Hat Game. Using covering codes, some optimal strategies for Hat Game will be introduced in both symmetric and asymmetric cases.

Keywords: Dimension, Maximal losing coalitions, Binary covering code, Symmetric case, Asymmetric case.



Faculty Of Mathematical Sciences

MSc Thesis in Applied Mathematics

On Covering Codes and Upper Bounds for the Dimension of Simple Games

By: Razieh Molaei

Supervisor:

Dr. Meysam Alishahi

Advisor:

Dr. Kazem Khashyarmanesh

December 2019