مجله تحقیق در عملیات در کاربردهای آن، سال هجدهم، شماره سوم (پیاپی ۷۰)، پاییز ۰۰، ص ص ۱۳-۱
شاپا ۷۲۸۶-۲۲۵۱

دو مدل برنامه‌ریزی خطی دودویی برای مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید

مریم اعتمادی[۱]، مهری باقریان[۲]*، حمیدرضا وزیری[۳]

[۱] دانشجوی دکتری گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

[۲] دانشیار، گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

[۳] دانشیار، گروه زیست شناسی، دانشکده علوم پایه، دانشگاه گیلان، رشت، ایران

رسید مقاله: ۲۲ آذر ۱۳۹۶

پذیرش مقاله: ۲۸ خرداد ۱۳۹۷

چکیده

مساله‌ی برهم‌گذاری هاپلوتایپ عبارت است از یافتن هاپلوتایپ‌های منشأ تعدادی قطعه که از روش‌های توالی‌یابی به‌دست آمده‌اند. درحالت دیپلوئید که مربوط به جانداران جفت کروموزومی مانند انسان است، درپی یافتن دو هاپلوتایپ هستیم که هر کدام از خوانش‌ها از یکی از دو هاپلوتایپ نشأت گرفته باشند. این مساله در حالت دیپلوئید بسیار مورد مطالعه قرار گرفته و به‌دلیل NP-hard بودن به خاطر وجود خطاهای اجتناب‌ناپذیر دستگاه‌های توالی‌یابی، روش‌های دقیق حل آن از مرتبه‌ی نمایی هستند. به‌همین دلیل روش‌های سریع‌تر ولی تقریبی زیادی نیز برای آن ارایه شده‌اند. درحالت تریپلوئید درپی یافتن سه هاپلوتایپ هستیم به‌طوری که هر یک از خوانش‌ها نشأت گرفته از یکی از سه هاپلوتایپ باشند. حالت تریپلوئید بسیار مشکل‌تر از حالت دیپلوئید بوده و با چالش محاسباتی بیشتری مواجه است. به‌همین دلیل محققان کمتر به این مساله پرداخته‌اند. در این مقاله دو مدل برنامه‌ریزی خطی دودویی برای این مساله در دو حالت وجود و عدم وجود اطلاعات ژنوتایپ ارایه و کارایی محاسباتی آن‌ها به‌کمک نرم‌افزار AIMMS روی داده‌های شبیه‌سازی‌شده مورد مطالعه قرار می‌گیرند. مدل‌های ارایه شده قابلیت تعمیم به پلوئیدی‌های بالاتر را نیز دارند.

کلمات کلیدی: برهم‌گذاری هاپلوتایپ، مدل برنامه‌ریزی خطی دودویی، تریپلوئید، روش‌های دقیق.

۱ مقدمه

همه‌ی موجودات زنده از سلول ساخته شده‌اند. درون هسته‌ی سلول رشته‌های طویلی قرار دارند که به‌آنها مولکول‌های DNA گفته می‌شود. این مولکول‌ها که اطلاعات ژنتیکی موجودات زنده را نگهداری می‌کنند، در زیر میکروسکوپ به صورت رشته‌های طویلی که کروموزوم نامیده می‌شود، قابل مشاهده‌اند. انسان‌ها موجوداتی

* عهده‌دار مکاتبات

آدرس الکترونیکی: mbagherian@guilan.ac.ir

[DOR: 20.1001.1.22517286.2021.18.3.2.9]

[DOI: 10.52547/jamlu.18.3.1]

دیپلوئید هستند یعنی به غیر ازکروموزوم‌های جنسی از سایر کروموزوم‌ها دو عدد دارند که یکی را از پدر و دیگری را از مادر به‌ارث می‌برند. تعداد کل کروموزوم‌های غیرجنسی در انسان چهل و چهار عدد یا بیست و دو جفت است. دو کروموزومی که تشکیل یک جفت می‌دهند بسیار مشابه ولی غیریکسان هستند و اصطلاحاً همولوگ نامیده می‌شوند. رایج‌ترین تفاوت در ژنوم افراد یک جمعیت، چندریختی تک نوکلئوتیدی یا اسنیپ (SNP[1]) است. درجایگاه‌های اسنیپ بیش از یک نوع باز بر روی ژنوم افراد مختلف جمعیت ملاحظه می‌شود که هر نوع متفاوت یک آلل نامیده می‌شود[۳-۱]. اگر یک جفت از آلل‌ها در یک مکان اسنیپ مقادیر یکسان داشته باشند، مکان اسنیپ هموزیگوت و درغیراین‌صورت هتروزیگوت است. مجموعه‌ی همه‌ی اسنیپ‌های واقع بر روی یک کروموزوم که با هم به نسل بعد منتقل می‌شوند، هاپلوتایپ نامیده می‌شود. به عبارت دیگر هاپلوتایپ متشکل از نواحی ژنی نزدیک به هم است که باهم به نسل بعد منتقل می‌شوند. تقریبا تمام چندریختی‌ها از دو آلل متفاوت تشکیل شده‌اند که می‌توان آلل با بیشترین فراوانی در یک مکان اسنیپ (آلل غالب) را با صفر و آلل دیگر (آلل مغلوب) را با یک کدگذاری نمود. با این کدگذاری یک هاپلوتایپ را می‌توان به صورت برداری از صفرها و یک‌ها نمایش داد[۴].

چون تعیین توالی هاپلوتایپ‌ها به روش‌های آزمایشگاهی کاری دشوار و پرهزینه است، روش‌های محاسباتی ابداع شده‌اند که این کار را انجام می‌دهند. ماشین‌های توالی‌یابی قادر به توالی‌یابی کل رشته‌های بلند DNA نیستند. این ماشین‌ها یک رشته بلند DNA را به قطعات کوتاه قابل توالی‌یابی برش می‌دهند و بعد از توالی‌یابی هر قطعه[2] با روی هم قراردادن آن‌ها توالی کل ژنوم را می‌یابند. تفکیک قطعات توالی‌یابی شده به دو یا چند دسته و روی هم‌گذاری آن‌ها برای بازسازی هاپلوتایپ‌ها مساله‌ی برهم‌گذاری هاپلوتایپ نامیده می‌شود. اگر قطعات کوتاه توالی‌یابی‌شده فاقد خطا باشند به سادگی می‌توان برای گونه‌های دیپلوئید دو هاپلوتایپ مجهول را تعیین نمود. اما وجود خطا باعث چالش محاسباتی قابل توجهی می‌شود، به‌طوری که ثابت‌شده که مساله در این حالت در رده‌ی مسایل NP-hard قرار می‌گیرد[۵]. تحقیقات زیادی روی مساله‌ی برهم‌گذاری هاپلوتایپ در حالت دیپلوئید انجام شده است [۱۲-۴،۶]. به‌طور کلی روش‌های مواجهه با مساله به دو دسته‌ی دقیق و تقریبی (ابتکاری) تقسیم‌بندی می‌شوند. روش‌های دقیق معمولاً مساله را به‌صورت یک مدل برنامه‌ریزی خطی دودویی فرمول‌بندی می‌کنند که برای حل آن‌ها می‌توان از حل‌کننده‌هایی نظیر گروبی[3] یا سیپلکس[4] بهره گرفت[۴،۶]. این روش‌ها برای مسایل واقعی که حجم داده‌ها بسیار زیاد است به زمان اجرای بسیار طولانی نیاز دارند. درمقابل روش‌های ابتکاری در زمان بسیار کوتاه‌تر یک جواب تقریبی برای مساله به دست می‌آورند که لزوماً بهینه نیست[۸،۷].

توابع هدف مختلفی نیز برای مساله‌ی برهم‌گذاری هاپلوتایپ معرفی شده‌اند که ازجمله می‌توان به کمترین تصحیح خطا (MEC[5]) [۵]، حذف کمترین تعداد اسنیپ (MSR[1])، حداقل حذف قطعه (MFR[2]) [۱۳]، حداکثر

[1] Single Nucleotide Polymorphism
[2] Fragment
[3] Gurobi
[4] CPLEX
[5] Minimum Error Correction

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

برش قطعه (MFC[3]) [۱۴] و افراز بهینه متعادل (BOP[4]) [۱۵] اشاره کرد. دراین مقاله روی تابع هدف کمترین تصحیح خطا متمرکز می‌شویم که تلاش می‌کند کمترین تعداد نوکلئوتیدها را در خوانش‌ها[5] بیابد که تغییر مقدار آن‌ها از صفر به یک یا برعکس تضادهای میان خوانش‌ها و هاپلوتایپ‌ها را برطرف کند.

برخی از محصولات کشاورزی مانند گندم، موز و سیب‌زمینی پلی‌پلوئید هستند، یعنی از هر کروموزوم بیش از دو عدد دارند. هاپلوتایپ‌ها اطلاعات کاملی درباره تفاوت‌های ژنتیکی درگونه‌های پلی‌پلوئید نتیجه می‌دهند اما مساله‌ی برهم‌گذاری هاپلوتایپ برای گونه‌های پلی‌پلوئید بسیار چالش برانگیزتر از گونه‌های دیپلوئید است. به‌همین دلیل تحقیقات بسیارکمی دراین زمینه انجام شده است. الگوریتم HapCompass [۱۶] مساله‌ی تعیین هاپلوتایپ‌ها را به مساله‌ی یافتن یک درخت فراگیر از گراف ساخته‌شده توسط خوانش‌های DNA تبدیل می‌کند. الگوریتم HapTree [۱۷]که از برآوردگر حداکثر خطا استفاده می‌کند برای کاهش پیچیدگی فرآیند محاسبات از یک استراتژی مشابه برنامه‌ریزی پویا استفاده می‌نماید. الگوریتم SDhaP [۱۸] مساله را به‌صورت یک مساله‌ی برنامه‌ریزی نیمه‌معین فرمول‌بندی نموده و با بهره‌گیری از ساختار خاص آن یعنی پایین بودن رتبه‌ی جواب اساسی آن، مساله را با دقت بالا و سرعت خوب حل می‌کند. روش آن‌ها را می‌توان هم درحالت دیپلوئید و هم درحالت تریپلوئید مورد استفاده قرار داد. الگوریتم H-POP [۱۹] مساله‌ی تعیین هاپلوتایپ‌ها را به‌صورت مساله‌ی دسته‌بندی خوانش‌ها مدل‌سازی می‌کند که (PBOP[6]) نامیده می‌شود. برای خوانش‌های توالی‌یابی‌شده از یک ژنوم k-ploid مدل سعی می‌کند که خوانش‌ها را به k گروه دسته‌بندی کند، به‌طوری که تفاوت میان خوانش‌های هم‌دسته مینیمم و تفاوت میان خوانش‌های دسته‌های متفاوت ماکزیمم شود. وقتی اطلاعات مربوط به ژنوتایپ دردسترس باشد، مدل به مساله‌ی PBOPG[7] تعمیم داده می‌شود. همه‌ی این مدل‌ها NP-hard هستند و برای حل آن‌ها دو الگوریتم ابتکاری HPOP و HPOPG به ترتیب براساس برنامه‌ریزی پویا و استراتژی محدودکردن جواب‌های میانی در هر تکرار ارایه شده است.

دراین مقاله دو مدل برنامه‌ریزی خطی دودویی برای مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید ارایه می‌دهیم که قابل تعمیم برای پلوئیدی‌های بالاتر نیز هستند. این‌ها اولین مدل‌های برنامه‌ریزی خطی دودویی هستند که برای مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید ارایه می‌شوند. هردو مدل مطرح شده درحالت وجود خطا در خوانش‌ها روی داده‌های شبیه‌سازی‌شده با طول‌ها و پوشش‌های مختلف اجرا شده و به لحاظ زمان حل توسط حل‌کننده‌ی AIMMS با یکدیگر مقایسه می‌شوند. اگرچه با افزایش پلوئیدی روش‌های دقیق بسیار کندتر از روش‌های ابتکاری مساله راحل می‌کنند، اما همچنان مدل‌های دقیق مورد توجه و ارزشمند هستند. زیرا دقت جواب‌های ارایه شده توسط آن‌ها بسیار بهتر است و نیز برای بررسی کیفیت جواب‌های ارایه‌شده توسط روش‌های ابتکاری و بهبود دقت جواب‌های آن‌ها می‌توان از روش‌های دقیق بهره گرفت.

---

[1] Minimum SNP Removal
[2] Minimum Fragment Removal
[3] Maximum Fragment Cut
[4] Balanced Optimal Partition
[5] reads
[6] Polyploid Balanced Optimal Partition
[7] Polyploid Balanced Optimal Partition with Genotype Constraints

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

۲ روش‌ها

ورودی مساله‌ی برهم‌گذاری هاپلوتایپ یک مجموعه از خوانش‌های توالی DNA مربوط به یک موجود است. توالی خوانش‌ها به ژنوم مرجع آرایه‌بندی می‌شوند تا جایگاه‌های اسنیپ به‌دست آیند. یک قطعه یک توالی خوانش آرایه‌بندی شده است که درآن مکان‌های حاوی بازهای یکسان حذف شده‌اند. جایگاه‌های هموزیگوت، کمکی به برهم‌گذاری هاپلوتایپ‌ها نمی‌کنند، چون قطعه‌ای که شامل آن آلل است نمی‌تواند به‌طور منحصربه‌فرد تعیین کند که آن آلل از کدام هاپلوتایپ نمونه‌برداری شده است. علاوه براین قطعاتی که شامل صفر یا یک اسنیپ هتروزیگوت هستند نیز برای برهم‌گذاری هاپلوتایپ مفید نیستند و می‌توان از آن‌ها صرف‌نظر کرد. فقط قطعاتی که شامل دو یا بیشتر اسنیپ هتروزیگوت هستند حاوی اطلاعات مفید برای برهم‌گذاری هاپلوتایپ‌ها هستند. قطعه‌ی iام که آن را r_i می‌نامیم به صورت یک بردار با مولفه‌های $\{-,0,1\}$ نمایش داده می‌شود. صفر و یک به‌ترتیب نمایش‌دهنده‌ی آلل‌های غالب و مغلوب دریک جایگاه اسنیپ و "–" نشان‌دهنده‌ی عدم وجود اطلاعات است یا به دلیل این که خوانش مورد نظر آن جایگاه اسنیپ را پوشش نمی‌دهد یا این‌که دستگاه توالی‌یابی نتوانسته باز آن جایگاه را بخواند. یک ماتریس $M_{m\times n}$ تعریف می‌کنیم که سطرهای آن متناظر با قطعات $r_1, r_2, \ldots, r_m$ و ستون‌های آن متناظر با اسنیپ‌های هتروزیگوت $s_1, s_2, \ldots, s_n$ هستند. چون قطعات در مقایسه با طول توالی‌های هاپلوتایپ نسبتاً کوتاه هستند ماتریس M تنک است، یعنی کسر بزرگی از درآیه‌های آن "–" است. هرقطعه‌ی r_i حداقل دو جایگاه از n جایگاه اسنیپ را پوشش می‌دهد. فاصله‌ی همینگ تعمیم‌یافته بین دو قطعه‌ی r_i و r_j با روابط (۱) و (۲) تعریف می‌شود.

$$d\left(r_i, r_j\right) = \sum_{k=1}^{n} d\left(r_{ik}, r_{jk}\right), \tag{۱}$$

که در آن

$$d\left(r_{ik}, r_{jk}\right) = \begin{cases} 1 & r_{ik} \neq r_{jk} \neq - \\ 0 & \text{درغیر این‌صورت} \end{cases} \tag{۲}$$

هدف مساله‌ی برهم‌گذاری هاپلوتایپ درحالت تریپلوئید تقسیم سطرهای ماتریس M به سه دسته است به‌طوری که از هر دسته یک هاپلوتایپ نتیجه شود. هاپلوتایپ‌ها را به‌صورت (۳)، (۴) و (۵) نمایش می‌دهیم.

$$h_1 = \left\{h_1^1, h_1^2, \ldots, h_1^n\right\}, \tag{۳}$$

$$h_2 = \left\{h_2^1, h_2^2, \ldots, h_2^n\right\}, \tag{۴}$$

$$h_3 = \left\{h_3^1, h_3^2, \ldots, h_3^n\right\}. \tag{۵}$$

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

درمساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید تحت تابع MEC می‌خواهیم هاپلوتایپ‌های $H=\{h_1,h_2,h_3\}$ را بیابیم به‌طوری که $MEC(M,H)$ که با روابط (۶) و (۷) تعریف می‌شود مینیمم گردد.

$$MEC(M,H)=\sum_{i=1}^{m} MEC(r_i,H), \quad (۶)$$

که درآن r_i سطر iام ماتریس M و

$$MEC(r_i,H)=min\{d(r_i,h_1),d(r_i,h_2),d(r_i,h_3)\}. \quad (۷)$$

برای نوشتن مدل ریاضی مساله دو حالت زیر را درنظر می‌گیریم.

۲-۱ حالتی که اطلاعات ژنوتایپ در دسترس است

دراین حالت تعداد یک‌های موجود در بین مولفه‌های نظیر هم در هاپلوتایپ‌ها را می‌دانیم که حداقل یک و حداکثر دو است، یعنی $h_1^j=h_2^j=h_3^j=0$ و $h_1^j=h_2^j=h_3^j=1$ رخ نمی‌دهند. در این حالت مدل‌های TILPG۱ و TILPG۲ را برای مساله‌ی برهم‌گذاری هاپلوتایپ ارایه می‌کنیم.

TILPG۱:

$$Min \sum_{i=1}^{m}\left(\sum_{j\in J_{i,0}} t_{ij}+\sum_{j\in J_{i,1}} u_{ij}\right) \quad (۸)$$

s.t.

$$h_1^j+y_1^i-1\le t_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,0} \quad (۹)$$

$$h_2^j+y_2^i-1\le t_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,0} \quad (۱۰)$$

$$1-\left(h_1^j+h_2^j\right)-y_1^i-y_2^i\le t_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,0} \quad (۱۱)$$

$$y_1^i-h_1^j\le u_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,1} \quad (۱۲)$$

$$y_2^i-h_2^j\le u_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,1} \quad (۱۳)$$

$$h_1^j+h_2^j-y_1^i-y_2^i\le u_{ij}, \qquad \forall i=1,\dots,m,\forall j\in J_{i,1} \quad (۱۴)$$

$$y_1^i+y_2^i\le 1, \qquad \forall i=1,\dots,m \quad (۱۵)$$

$$h_1^j+h_2^j\le 1, \qquad \forall j=1,\dots,n \quad (۱۶)$$

$$y_1^i,y_2^i,h_1^j,h_2^j,t_{ij},u_{ij}\in\{0,1\}, \qquad \forall i=1,\dots,m,\forall j=1,\dots,n. \quad (۱۷)$$

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

$J_{i,0}$ ($J_{i,1}$) مجموعه‌ی ستون‌هایی در ماتریس M است که درسطر i ام مقدار صفر (یک) دارند. محدودیت‌های (۹)، (۱۰) و (۱۱) کران پایینی مناسب برای t_{ij} و محدودیت‌های (۱۲)، (۱۳) و (۱۴) کران پایینی مناسب برای u_{ij} تولید می‌کنند. همچنین محدودیت‌های (۱۵) به این معنا هستند که فقط حالت‌های (الف)، (ب) و (پ) زیر می‌توانند رخ دهند.

(الف) $y_1^i=1$ و $y_2^i=0$ اگروتنهااگر سطر i ام به‌هاپلوتایپ h_1 تخصیص داده شود.

(ب) $y_1^i=0$ و $y_2^i=1$ اگروتنهااگر سطر i ام به هاپلوتایپ h_2 تخصیص داده شود.

(پ) $y_1^i=0$ و $y_2^i=0$ اگروتنهااگر سطر i ام به هاپلوتایپ h_3 تخصیص داده شود.

محدودیت‌های (۱۶) تضمین می‌کنند که حالت $h_1^j=h_2^j=1$ امکان وقوع ندارد. h_1^j و h_2^j به‌ترتیب نشان‌دهنده‌ی بیت j ام هاپلوتایپ‌های h_1 و h_2 هستند که می‌توانند صفر یا یک باشند. درحالتی که اطلاعات ژنوتایپ را داشته باشیم به‌جای h_3^j می‌توانیم بنویسیم $1-\left(h_1^j+h_2^j\right)$. متغیرهای باینری t_{ij} و u_{ij} متضاد بودن (جریمه) یا نبودن بیت j ام سطر i ام را با هاپلوتایپی که به آن تخصیص‌یافته است نشان می‌دهند و مقدار آن‌ها یک است اگر و فقط اگر بیت j ام سطر i ام با بیت j ام هاپلوتایپی که به آن تخصیص‌یافته در تضاد باشد. درغیر این‌صورت این متغیرها مقدار صفر خواهند داشت.

حال نشان می‌دهیم که با مدل TILPG۱ سطرهای ماتریس ورودی را به سه دسته تقسیم می‌کنیم به طوری که سطرهای موجود در دسته‌ی هر هاپلوتایپ با آن هاپلوتایپ دارای کم‌ترین تعداد تضاد هستند. در جدول ۱ همه‌ی حالت‌هایی که می‌تواند برای متغیر t_{ij} رخ دهد در حالتی که $j\in J_{i,0}$ داده شده است. نشان می‌دهیم که برای هر $i\in\{1,\cdots,m\}$ و هر $j\in J_{i,0}$، $t_{ij}=1$ اگروتنهااگر $h_1^j=1$ یا $y_1^i=1, y_2^i=0$، $h_3^j=1-\left(h_1^j+h_2^j\right)=1$ یا $y_1^i=0, y_2^i=1, h_4^j=1$ یا $y_1^i=0, y_2^i=0$.

$h_1^j=1$ و $y_1^i=1, y_2^i=0$ با هم به این معنا هستند که سطر i ام به هاپلوتایپ h_1 تخصیص داده شده و درآیه‌ی j ام آن صفر است ($j\in J_{i,0}$) و با درآیه‌ی j ام h_1 متفاوت است ($h_1^j=1$). $y_1^i=0, y_2^i=1, h_4^j=1$ با هم به این معنا هستند که سطر i ام به هاپلوتایپ h_2 تخصیص داده شده اما درآیه‌ی j ام آن با درآیه‌ی j ام h_2 متفاوت است. $1=1-\left(h_1^j+h_2^j\right)$ $y_1^i=0, y_2^i=0, h_3^j=1$ به این معنا هستند که سطر i ام به هاپلوتایپ h_4 تخصیص داده شده اما درآیه‌ی j ام آن با درآیه‌ی j ام h_3 متفاوت است. متغیر t_{ij} در تابع هدف دارای ضریب مثبت است و از آن جایی که تابع هدف از نوع مینیمم است و t_{ij} نیز دودویی بوده و فقط می‌تواند صفر یا یک باشد، در حالتی که کران پایینی مثبت نداشته باشد مقدار صفر خواهد داشت. با توجه به محدودیت‌های (۱۱)-(۹)، t_{ij} دارای سه کران پایینی $h_1^j+y_1^i-1$، $h_2^j+y_2^i-1$ و $1-\left(h_1^j+h_2^j\right)-y_1^i-y_2^i$ است. با تغییر مقادیر متغیرها این سه محدودیت کران‌های پایینی متفاوتی را برای t_{ij} ایجاد می‌کنند. برای مثال در حالتی که $y_1^i=1$ و $y_2^i=0$ داریم:

$$h_1^j+y_1^i-1=h_1^j\le t_{ij} \qquad (18)$$

$$h_2^j+y_2^i-1=h_2^j-1\le t_{ij} \qquad (19)$$

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

$$۱-\left(h_۱^j+h_۲^j\right)-y_۱^i-y_۲^i=-\left(h_۱^j+h_۲^j\right)\le t_{ij}\ . \qquad (۲۰)$$

در بین سه کران پایینی به دست آمده از سه محدودیت (۱۸)، (۱۹) و (۲۰) برای متغیر t_{ij}، $h_۱^j$ از بقیه بزرگ‌تر است و لذا t_{ij} مساوی $h_۱^j$ خواهد شد. در نتیجه اگر $h_۱^j=۱$ ،$y_۱^i=۱, y_۲^i=۰$ آنگاه $t_{ij}=۱$ است. یعنی در حالتی که $j\in J_{i,۰}$ (بیت j ام سطر i ام صفر است) و $h_۱^j=۱$ (بیت j ام هاپلوتایپ $h_۱$ یک است) و $y_۱^i=۱, y_۲^i=۰$ (سطر i ام به هاپلوتایپ $h_۱$ تخصیص‌داده شده) از آن جایی که بیت‌های j ام سطر i ام و هاپلوتایپ $h_۱$ یکسان نیستند ولی این سطر به $h_۱$ تخصیص‌یافته، جریمه‌ای به اندازه‌ی یک واحد به تابع هدف اضافه می‌شود یعنی $t_{ij}=۱$ می‌گردد. به طریق مشابه در حالتی که $y_۱^i=۰, y_۲^i=۱, h_۲^j=۱$ یا $y_۱^i=۰, y_۲^i=۰, h_۳^j=۱-\left(h_۱^j+h_۲^j\right)=۱$ می‌توان نشان داد که $t_{ij}=۱$ می‌شود. همچنین برای هر $i\in\{۱,\cdots,m\}$ و هر $j\in J_{i,۱}$، $u_{ij}=۱$ اگر و تنها اگر $y_۱^i=۱, y_۲^i=۰, ۱-h_۱^j=۱$ یا $y_۱^i=۰, y_۲^i=۱, ۱-h_۲^j=۱$ یا $y_۱^i=۰, y_۲^i=۰, ۱-h_۳^j=h_۱^j+h_۲^j=۱$. اثبات این مطلب مشابه حالت $j\in J_{i,۰}$ است با این تفاوت که از جدول ۲ استفاده می‌شود.

جدول ۱. مقادیر t_{ij} برای $j\in J_{i,۰}$

$h_۱^j$	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱
$h_۲^j$	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱
$h_۳^j$	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱
$y_۱^i$	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰
$y_۲^i$	۰	۰	۰	۰	۰	۰	۰	۰	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰
t_{ij}	۰	۱	۰	۰	۱	۱	۰	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱

جدول ۲. مقادیر u_{ij} برای $j\in J_{i,۱}$

$h_۱^j$	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱
$h_۲^j$	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱
$h_۳^j$	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱
$y_۱^i$	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰
$y_۲^i$	۰	۰	۰	۰	۰	۰	۰	۰	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰
u_{ij}	۱	۰	۱	۱	۰	۰	۱	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۱	۰	۱	۰	۰	۰

مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید و با فرض در دسترس بودن داده‌های ژنوتایپ را به صورت دیگری نیز می‌توان فرمول‌بندی نمود که در آن متغیرهای باینری t_{ij} و u_{ij} خود نشان‌دهنده‌ی جریمه نیستند بلکه از آن‌ها برای تولید جریمه در تابع هدف استفاده می‌شود. مدل TILPG۲ مساله‌ی برهم‌گذاری هاپلوتایپ را در حالت تریپلوئید و با فرض در دسترس بودن داده‌های ژنوتایپ فرمول‌بندی می‌کند.

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

$TILPG۲$:

$$Min\sum_{i=۱}^{m}\left(\begin{array}{l}\sum_{j\in J_{i,۰}}\left[۱-\left(h_{۱}^{j}+h_{۲}^{j}\right)-\left(y_{۱}^{i}+y_{۲}^{i}\right)+t_{ij}+u_{ij}\right]\\ +\sum_{j\in J_{i,۱}}\left[\left(h_{۱}^{j}+h_{۲}^{j}\right)-\left(y_{۱}^{i}+y_{۲}^{i}\right)+t_{ij}+u_{ij}\right]\end{array}\right) \quad (۲۱)$$

$s.t.$

$$۲h_{۱}^{j}+h_{۲}^{j}+۲y_{۱}^{i}-۲\le t_{ij}+u_{ij}\ , \qquad \forall i=۱,\ldots,m,\forall j\in J_{i,۰} \quad (۲۲)$$

$$h_{۱}^{j}+۲h_{۲}^{j}+۲y_{۲}^{i}-۲\le t_{ij}+u_{ij}\ , \qquad \forall i=۱,\ldots,m,\forall j\in J_{i,۰} \quad (۲۳)$$

$$-۲h_{۱}^{j}-h_{۲}^{j}+۲y_{۱}^{i}\le t_{ij}+u_{ij}, \qquad \forall i=۱,\ldots,m,\forall j\in J_{i,۱} \quad (۲۴)$$

$$-h_{۱}^{j}-۲h_{۲}^{j}+۲y_{۲}^{i}\le t_{ij}+u_{ij}, \qquad \forall i=۱,\ldots,m,\forall j\in J_{i,۱} \quad (۲۵)$$

$$y_{۱}^{i}+y_{۲}^{i}\le ۱, \qquad \forall i=۱,\ldots,m \quad (۲۶)$$

$$h_{۱}^{j}+h_{۲}^{j}\le ۱, \qquad \forall j=۱,\ldots,n \quad (۲۷)$$

$$y_{۱}^{i},y_{۲}^{i},h_{۱}^{j},h_{۲}^{j},t_{ij},u_{ij}\in\{۰,۱\}, \qquad \forall i=۱,\ldots,m,\forall j=۱,\ldots,n. \quad (۲۸)$$

متغیرهای مدل $TILPG۲$ مانند مدل $TILPG۱$ تعریف می‌شوند. محدودیت‌های (۲۲)، (۲۳)، (۲۴) و (۲۵) کران‌های پایینی مناسب برای $t_{ij}+u_{ij}$ تولید می‌کنند. محدودیت‌های (۲۶) نشان می‌دهند که حالت $y_{۱}^{i}=y_{۲}^{i}=۱$ امکان وقوع ندارد. محدودیت‌های (۲۷) تضمین می‌کنند که حالت $h_{۱}^{j}=h_{۲}^{j}=۱$ امکان وقوع ندارد. همچنین درحالتی که اطلاعات ژنوتایپ را داشته باشیم به‌جای $h_{۳}^{j}$ می‌توانیم بنویسیم $۱-\left(h_{۱}^{j}+h_{۲}^{j}\right)$. اثبات این‌که مدل $TILPG۲$ نیز مانند مدل $TILPG۱$ به درستی سطرهای ماتریس ورودی را به سه دسته تقسیم می‌کند به طوری که سطرهای موجود در هر دسته با هاپلوتایپ نظیر آن کم‌ترین تضاد را داشته باشند به طریق مشابه با استفاده از جداول ۳ و ۴ انجام می‌شود.

برای هر $i\in\{۱,\cdots,m\}$ و هر $j\in J_{i,۰}$، $۱-\left(h_{۱}^{j}+h_{۲}^{j}\right)-\left(y_{۱}^{i}+y_{۲}^{i}\right)+t_{ij}+u_{ij}=۱$ اگر و تنها اگر $y_{۱}^{i}=۱, y_{۲}^{i}=۰, h_{۱}^{j}=۱$ یا $y_{۱}^{i}=۰, y_{۲}^{i}=۱, h_{۲}^{j}=۱$ یا $y_{۱}^{i}=۰, y_{۲}^{i}=۰, h_{۳}^{j}=۱$. تعریف‌های این سه حالت مشابه تعریف‌های گفته شده برای آن‌ها در مدل $TILPG۱$ است. در جدول ۳ همه‌ی مقادیر ممکن برای متغیرهای $y_{۱}^{i}, y_{۲}^{i}, h_{۱}^{j}, h_{۲}^{j}, h_{۳}^{j}$ در نظر گرفته شده‌اند. عبارت $t_{ij}+u_{ij}$ در تابع هدف دارای ضریب مثبت بوده و تابع هدف از نوع مینیمم است. با توجه به باینری بودن متغیرهای t_{ij} و u_{ij} کمترین مقدار عبارت $t_{ij}+u_{ij}$ صفر است مگر آن که این عبارت کران پایینی مثبتی داشته باشد. در محدودیت های مدل $TILPG۲$ در حالت $j\in J_{i,۰}$ دو کران پایین $۲h_{۱}^{j}+h_{۲}^{j}+۲y_{۱}^{i}-۲$ و $h_{۱}^{j}+۲h_{۲}^{j}+۲y_{۲}^{i}-۲$ برای این عبارت وجود دارد. برای مثال در حالتی که $y_{۱}^{i}=۱, y_{۲}^{i}=۰$ داریم:

$$۲h_{۱}^{j}+h_{۲}^{j}+۲y_{۱}^{i}-۲=۲h_{۱}^{j}+h_{۲}^{j}\le t_{ij}+u_{ij} \quad (۲۹)$$

$$h_{۱}^{j}+۲h_{۲}^{j}+۲y_{۲}^{i}-۲=h_{۱}^{j}+۲h_{۲}^{j}-۲\le t_{ij}+u_{ij}. \quad (۳۰)$$

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

در بین دو کران پایینی به دست آمده از دو محدودیت (۲۹) و (۳۰)، $۲h_۱^j + h_۲^j$ بزرگ‌تر است، در نتیجه $t_{ij} + u_{ij}$ مساوی $۲h_۱^j + h_۲^j$ می‌شود و لذا مقدار تابع هدف برای $j \in J_{i,۰}$ مساوی $h_۱^j$ شده، بنابراین وقتی $y_۱^i = ۱, y_۲^i = ۰, h_۱^j = ۱$ مقدار هدف نیز یک می‌گردد. به طریق مشابه می‌توان نشان داد در حالتی که $y_۱^i = ۰, y_۲^i = ۱, h_۲^j = ۱$ یا $y_۱^i = ۰ , y_۲^i = ۰ \quad h_۳^j = ۱ - \left(h_۱^j + h_۲^j\right) = ۱$ مقدار تابع هدف نیز مساوی یک می‌گردد.

همچنین برای هر $i \in \{۱, \cdots, m\}$ و هر $j \in J_{i,۱}$، $\left(h_۱^j + h_۲^j\right) - \left(y_۱^i + y_۲^i\right) + t_{ij} + u_{ij} = ۱$ اگر وتنها اگر $y_۱^i = ۱ , y_۲^i = ۰, ۱ - h_۱^j = ۱$ یا $y_۱^i = ۰, y_۲^i = ۱, ۱ - h_۲^j = ۱$ یا $y_۱^i = ۰, y_۲^i = ۰, ۱ - h_۳^j = \left(h_۱^j + h_۲^j\right) = ۱$ اثبات این مطلب مشابه حالت $j \in J_{i,۰}$ است با این تفاوت که از جدول ۴ استفاده می‌شود.

جدول ۳. مقادیر $۱ - \left(h_۱^j + h_۲^j\right) - \left(y_۱^i + y_۲^i\right) + t_{ij} + u_{ij}$ برای $j \in J_{i,۰}$

$h_۱^j$	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱
$h_۲^j$	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱
$h_۳^j = ۱ - \left(h_۱^j + h_۲^j\right)$	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱
$y_۱^i$	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰
$y_۲^i$	۰	۰	۰	۰	۰	۰	۰	۰	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰
$۱ - \left(h_۱^j + h_۲^j\right) - \left(y_۱^i + y_۲^i\right) + t_{ij} + u_{ij}$	۰	۱	۰	۰	۱	۱	۰	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱

جدول ٤. مقادیر $\left(h_۱^j + h_۲^j\right) - \left(y_۱^i + y_۲^i\right) + t_{ij} + u_{ij}$ برای $j \in J_{i,۱}$

$h_۱^j$	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۰	۱
$h_۲^j$	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱	۰	۰	۱	۰	۱	۰	۱	۱
$h_۳^j = ۱ - \left(h_۱^j + h_۲^j\right)$	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱	۰	۰	۰	۱	۰	۱	۱	۱
$y_۱^i$	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰
$y_۲^i$	۰	۰	۰	۰	۰	۰	۰	۰	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰
$\left(h_۱^j + h_۲^j\right) - \left(y_۱^i + y_۲^i\right) + t_{ij} + u_{ij}$	۱	۰	۱	۱	۰	۰	۱	۰	۱	۱	۰	۱	۰	۱	۰	۰	۱	۱	۱	۰	۱	۰	۰	۰

۲-۲ حالتی که اطلاعات ژنوتایپ در دسترس نیست

دراین حالت تعداد یک‌های موجود در بین مولفه‌های نظیر هم در هاپلوتایپ‌ها را نمی‌دانیم و حالت‌های $h_۱^j = h_۲^j = h_۳^j = ۰$ و $h_۱^j = h_۲^j = h_۳^j = ۱$ نیز ممکن است رخ دهند. تعمیم مدل‌های TILPG۱ و TILPG۲ را می‌توان به‌ترتیب به‌صورت TILPNG۱ و TILPNG۲ نوشت.

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

*TILPNG*۱:

$$Min \sum_{i=۱}^{m} \left(\sum_{j \in J_{i,۰}} t_{ij} + \sum_{j \in J_{i,۱}} u_{ij} \right) \quad (۳۱)$$

s.t.

$$h_{۱}^{j} + y_{۱}^{i} - ۱ \le t_{ij} \quad , \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۰} \quad (۳۲)$$

$$h_{۲}^{j} + y_{۲}^{i} - ۱ \le t_{ij} \quad , \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۰} \quad (۳۳)$$

$$h_{۳}^{j} - y_{۱}^{i} - y_{۲}^{i} \le t_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۰} \quad (۳۴)$$

$$y_{۱}^{i} - h_{۱}^{j} \le u_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۱} \quad (۳۵)$$

$$y_{۲}^{i} - h_{۲}^{j} \le u_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۱} \quad (۳۶)$$

$$۱ - (y_{۱}^{i} + y_{۲}^{i}) - h_{۳}^{j} \le u_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۱} \quad (۳۷)$$

$$y_{۱}^{i} + y_{۲}^{i} \le ۱, \qquad \forall i = ۱,\ldots,m \quad (۳۸)$$

$$y_{۱}^{i}, y_{۲}^{i}, h_{۱}^{j}, h_{۲}^{j}, h_{۳}^{j}, t_{ij}, u_{ij} \in \{۰,۱\}, \qquad \forall i = ۱,\ldots,m, \forall j = ۱,\ldots,n \quad (۳۹)$$

و

*TILPNG*۲:

$$Min \sum_{i=۱}^{m} \left(\sum_{j \in J_{i,۰}} \left[h_{۳}^{j} - \left(y_{۱}^{i} + y_{۲}^{i} \right) + t_{ij} + u_{ij} \right] + \sum_{j \in J_{i,۱}} \left[۱ - h_{۳}^{j} - \left(y_{۱}^{i} + y_{۲}^{i} \right) + t_{ij} + u_{ij} \right] \right) \quad (۴۰)$$

s.t.

$$h_{۱}^{j} - h_{۳}^{j} + ۲y_{۱}^{i} - ۱ \le t_{ij} + u_{ij} \quad , \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۰} \quad (۴۱)$$

$$h_{۲}^{j} - h_{۳}^{j} + ۲y_{۲}^{i} - ۱ \le t_{ij} + u_{ij} \quad , \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۰} \quad (۴۲)$$

$$-h_{۱}^{j} + h_{۳}^{j} + ۲y_{۱}^{i} - ۱ \le t_{ij} + u_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۱} \quad (۴۳)$$

$$-h_{۲}^{j} + h_{۳}^{j} + ۲y_{۲}^{i} - ۱ \le t_{ij} + u_{ij}, \qquad \forall i = ۱,\ldots,m, \forall j \in J_{i,۱} \quad (۴۴)$$

$$y_{۱}^{i} + y_{۲}^{i} \le ۱, \qquad \forall i = ۱,\ldots,m \quad (۴۵)$$

$$y_{۱}^{i}, y_{۲}^{i}, h_{۱}^{j}, h_{۲}^{j}, h_{۳}^{j}, t_{ij}, u_{ij} \in \{۰,۱\}, \qquad \forall i = ۱,\ldots,m, \forall j = ۱,\ldots,n. \quad (۴۶)$$

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

تعاریف متغیرهای مدل‌های TILPNG۱ وTILPNG۲ همانند مدل‌هایTILPG۱ وTILPG۲ است. یک متغیر جدید $h_۴^j$ در مدل‌های TILPNG۱ وTILPNG۲ وجود دارد که جایگزین $۱-\left(h_۱^j+h_۲^j\right)$ در مدل‌های TILPG۱ وTILPG۲ شده است. همچنین اثبات درستی مدل‌هایTILPNG۱ و TILPNG۲ مشابه مدل‌های TILPG۱ و TILPG۲ با استفاده از جداول ۱، ۲، ۳ و ۴ انجام می‌شود با این تفاوت که در این مدل‌ها حالت‌های $h_۱^j=h_۲^j=h_۴^j=۰$ و $h_۱^j=h_۲^j=h_۴^j=۱$ را نیز در جداول در نظر می‌گیریم.

۳ نتایج محاسباتی

از آن‌جایی که این مدل‌های برنامه‌ریزی خطی دودویی اولین مدل‌هایی هستند که برای مساله‌ی برهم‌گذاری هاپلوتایپ درحالت تریپلوئید طراحی شده‌اند، مدل‌های ارایه‌شده را از نظر زمان حل توسط یک حل‌کننده‌ی سریع مسایل برنامه‌ریزی خطی دودویی با هم مقایسه کردیم. بدین منظور داده‌هایی شبیه‌سازی کرده و عملکرد دو مدل را روی یک سیستم با مشخصات ۱۶ گیگابایت رم و پردازشگر i۷-۴۷۹۰ مورد ارزیابی قرار دادیم. در شبیه‌سازی داده‌ها سه پارامتر لحاظ شدند. طول هاپلوتایپ، نرخ خطا و نرخ پوشش که به ترتیب با l، e و c نمایش داده می‌شوند. ما با مقادیر $e=۰/۱$ ،$l=۱۰,۲۰,۳۰,۴۰$ و $c=۳,۴,۵$ داده تولید نمودیم. نرخ پوشش بیانگر این است که حداکثر تعداد صفرها و یک‌ها در هر ستون ماتریس ورودی ۳c است. برای هر ترکیب از سه پارامتر ۱۰ ماتریس نمونه تولید و متوسط زمان حل و MEC آن‌ها را محاسبه نمودیم. در جدول ۵ زمان اجرا (برحسب ثانیه) و MEC برای مدل‌های TILPG۱ وTILPG۲ توسط حل‌کننده‌ی AIMMS آورده شده است.

جدول ۵. متوسط زمان اجرا و MEC روی ۱۰ ماتریس با $l=۱۰,۲۰,۳۰,۴۰$ ،$c=۳,۴,۵$ و $e=۰/۱$ در حالت در دسترس بودن اطلاعات ژنوتایپ

	مدل TILPG۱		مدل TILPG۲	
(l,c)	MEC	زمان حل (ثانیه)	MEC	زمان حل (ثانیه)
(۱۰,۳)	۱/۹	۰/۰۲	۱/۹	۰/۰۷
(۱۰,۴)	۳/۳	۰/۰۴۶	۳/۳	۰/۱۵۲
(۱۰,۵)	۵/۲	۰/۰۸	۵/۲	۰/۳۰۶
(۲۰,۳)	۱۴/۱	۰/۲۸۷	۱۴/۱	۴/۵۱۲
(۲۰,۴)	۱۸/۲	۱/۰۲۶	۱۸/۲	۸/۴۲۲
(۲۰,۵)	۲۵/۷	۱/۳۶۴	۲۵/۷	۳۱/۲۵۵
(۳۰,۳)	۲۴/۴	۲/۴۱	۲۴/۴	۲۰/۱۰۱
(۳۰,۴)	۳۱/۳	۷/۵۶۸	۳۱/۳	۵۲/۴۹۱
(۳۰,۵)	۴۲/۲	۲۸/۹۳۶	۴۲/۲	۲۴۸/۸۰۹
(۴۰,۳)	۳۲/۸	۳/۱۲۳	۳۲/۸	۲۵/۴۵۵
(۴۰,۴)	۴۷/۶	۳۱/۳۷۸	۴۷/۶	۱۶۳/۴۲۲
(۴۰,۵)	۵۷/۹	۲۷۲/۰۶۹	۵۷/۹	۲۹۷۶/۸۴۳

[DOR: 20.1001.1.22517286.2021.18.3.2.9]
[DOI: 10.52547/jamlu.18.3.1]

همان‌طور که ملاحظه می‌شود مدل TILPG۱ نسبت به مدل TILPG۲ در زمان کوتاه‌تری به جواب می‌رسد. علاوه براین با افزایش l و c زمان حل مدل‌ها افزایش می‌یابد و تفاوت زمان حل دو مدل بیشتر نمایان می‌گردد.

در جدول ۶ مدل‌های TILPNG۱ و TILPNG۲ که مربوط به حالتی هستند که اطلاعات ژنوتایپ در دسترس نیست از نظر زمان حل با هم مورد مقایسه قرار گرفته‌اند. سیستم برای حل ۱۰ ماتریس با طول و نرخ پوشش $l=۴۰$ و $c=۵$ با کمبود رم مواجه می‌شود و لذا محاسبه متوسط زمان حل امکان‌پذیر نیست. اما با توجه به دقیق بودن هر دو مدل انتظار می‌رود که مقدار MEC مدل TILPNG۲ نیز برابر با مقدار MEC مدل TILPNG۱ یعنی ۵۱/۹ باشد.

جدول ٦. متوسط زمان اجرا و MEC روی ۱۰ ماتریس با $l=۱۰,۲۰,۳۰,۴۰$ و $c=۳,۴,۵$ و $e=۰/۱$ در حالت عدم دسترسی به اطلاعات ژنوتایپ

	مدل TILPNG۱		مدل TILPNG۲	
(l,c)	MEC	زمان حل (ثانیه)	MEC	زمان حل (ثانیه)
(۱۰,۳)	۰/۷	۰/۰۷۵	۰/۷	۰/۰۹۹
(۱۰,۴)	۱/۹	۰/۱۳۶	۱/۹	۰/۱۶۷
(۱۰,۵)	۳/۹	۰/۲۶۸	۳/۹	۰/۵۶۴
(۲۰,۳)	۱۰/۳	۲/۹۸۸	۱۰/۳	۴/۷۲۳
(۲۰,۴)	۱۴/۲	۸/۴۹۶	۱۴/۲	۱۲/۲۳۳
(۲۰,۵)	۲۰/۳	۳۵/۲۳	۲۰/۳	۱۱۱/۹۴۶
(۳۰,۳)	۱۸/۳	۱۲/۴۸۹	۱۸/۳	۲۵/۰۴۳
(۳۰,۴)	۲۷/۹	۱۱۴/۵۲۸	۲۷/۹	۳۷۸/۰۵۷
(۳۰,۵)	۳۵/۸	۷۰۱/۵۶۶	۳۵/۸	۴۲۶۹/۷۹۲
(۴۰,۳)	۲۴/۸	۳۲/۵۱۴	۲۴/۸	۸۹/۸۵۱
(۴۰,۴)	۳۶/۳	۲۱۲/۲۹	۳۶/۳	۶۲۹/۲۳۹
(۴۰,۵)	۵۱/۹	۹۰۲۸/۱۶۴	۵۱/۹	–

٤ بحث

در این مقاله دو مدل برنامه‌ریزی خطی دودویی برای مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید و با فرض وجود و عدم وجود اطلاعات ژنوتایپ طراحی شدند. عملکرد مدل‌ها از نظر زمان اجرا روی داده‌های شبیه‌سازی‌شده مورد بررسی قرار گرفتند. نتایج محاسباتی نشان داد که یکی از مدل‌ها بسیار سریع‌تر از دیگری توسط حل‌کننده‌های سریع مسایل برنامه‌ریزی خطی دودویی حل می‌شود. این مدل‌ها اولین مدل‌های برنامه‌ریزی خطی دودویی هستند که برای حل مساله‌ی برهم‌گذاری هاپلوتایپ در حالت تریپلوئید طراحی می‌شوند. از این مدل‌ها می‌توان برای ارزیابی جواب حاصل از مدل‌های ابتکاری و نیز بهبود آن‌ها بهره گرفت.

منابع

[1] Alberts, B., Bray, D., Hopkin, K., Johnson, A. D., Lewis, J., Raff, M., Walter, P. (2015). Essential cell biology. Garland Science.

[2] Alberts, B., Johnson, A., Morgan, D., Raff, M., Roberts, R., and Walter, P., (2015), Molecular biology of the cell, Garland Science, 6th edition, USA.
[3] Watson, J. D., Baker, T. A., Gann, A., Levine, M., Losick, R., (2014), Molecular biology of the gene, Cold spring laboratory press, 7th edition, USA.
[4] Chen, Z.-Z., Deng, F., Wang, L. (2013). Exact algorithms for haplotype assembly from whole-genome sequence data. Bioinformatics, 29, 1938-45.
[5] Lippert, R., Schwartz, R., Lancia, G., & Istrail, S. (2002). Algorithmic strategies for the single nucleotide polymorphism haplotype assembly problem. Briefings in bioinformatics, 3(1), 23-31.
[6] Chen, Z.-Z., Deng, F., Shen, C., Wang, Y., Wang, L. (2016). Better ILP-based approaches to haplotype assembly. Journal of Computational Biology, 23, 537-52.
[7] Levy, S., Sutton, G., Ng, P.C., Feuk, L., Halpern, A.L., Walenz, B.P., Axelrod, N., Huang, J., Kirkness, E.F., Denisov, G., Lin, Y. (2007). The diploid genome sequence of an individual human. PLoS biology, 5, 254.
[8] Panconesi, A., Sozio, M. (2004). Fast hare: A fast heuristic for single individual snp haplotype reconstruction. In:International Workshop on Algorithms in Bioinformatics, pp. 266-277. Springer.
[9] He, D., Choi, A., Pipatsrisawat, K., Darwiche, A., Eskin, E. (2010). Optimal algorithms for haplotype assembly from whole-genome sequence data. Bioinformatics, 26, 183-90.
[10] Bansal, V., Bafna, V. (2008). Hapcut: an efficient and accurate algorithm for the haplotype assembly problem. Bioinformatics, 24, 153-9.
[11] Bansal, V., Halpern, A.L., Axelrod, N., Bafna, V. (2008). An mcmc algorithm for haplotype assembly whole-genome sequence data. Genome research, 18, 1336-46.
[12] Mousavi, S.R., Khodadadi, I., Falsafain, H., Nadimi, R., Ghadiri, N. (2014). Maximum likelihood model based on minor allele frequencies and weighted max-sat formulation for haplotype assembly. Journal of theoretical biology, 350, 49-56.
[13] Lancia, G., Bafna, V., Istrail, S., Lippert, R., & Schwartz, R. (2001). SNPs problems, complexity, and algorithms. In ESA (Vol. 1, pp. 182-193).
[14] Duitama, J., Huebsch, T., McEwen, G., Suk, E. K., & Hoehe, M. R. (2010). ReFHap: a reliable and fast algorithm for single individual haplotyping. In Proceedings of the First ACM International Conference on Bioinformatics and Computational Biology (pp. 160-169). ACM.
[15] Xie, M., Wang, J., & Jiang, T. (2012). A fast and accurate algorithm for single individual haplotyping. BMC systems biology, 6(2), S8.
[16] Aguiar, D., & Istrail, S. (2013). Haplotype assembly in polyploid genomes and identical by descent shared tracts. Bioinformatics, 29(13), i352-i360.
[17] Berger, E., Yorukoglu, D., Peng, J., & Berger, B. (2014). Haptree: A novel bayesian framework for single individual polyplotyping using ngs data. PLoS computational biology, 10(3), e1003502.
[18] Das, S., & Vikalo, H. (2015). SDhaP: haplotype assembly for diploids and polyploids via semi-definite programming. BMC genomics, 16(1), 260.
[19] Xie, M., Wu, Q., Wang, J., & Jiang, T. (2016). H-PoP and H-PoPG: heuristic partitioning algorithms for single individual haplotyping of polyploids. Bioinformatics, 32(24), 3735-3744.