

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

مریم اعتمادی<sup>۱</sup>، مهری باقریان<sup>۲\*</sup>، حمیدرضا وزیری<sup>۳</sup>

<sup>۱</sup> دانشجوی دکتری گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

<sup>۲</sup> دانشیار، گروه ریاضی کاربردی، دانشکده علوم ریاضی، دانشگاه گیلان، رشت، ایران

<sup>۳</sup> دانشیار، گروه زیست‌شناسی، 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. در حالت دیپلوئید که مربوط به جانداران جفت کروموزومی مانند انسان است، در پی 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 به دلیل NP-hard بودن به خاطر وجود خطاهای اجتناب‌ناپذیر دستگاه‌های توالی‌یابی، روش‌های دقیق حل آن از مرتبه‌ی نمایی هستند. به همین دلیل روش‌های سریع‌تر ولی تقریبی زیادی نیز برای آن ارایه شده‌اند. در حالت تریپلوئید در پی 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. به همین 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 به کمک 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 طولی قرار دارند که به آنها مولکول‌های DNA گفته می‌شود. این مولکول‌ها که اطلاعات ژنتیکی موجودات زنده را نگهداری می‌کنند، در زیر میکروسکوپ به صورت رشته‌های طولی که کروموزوم نامیده می‌شود، قابل مشاهده‌اند. انسان‌ها موجوداتی

\* عهده‌دار مکاتبات

آدرس الکترونیکی: mbagherian@guilan.ac.ir

دیپلوئید هستند یعنی به غیر از کروموزوم‌های جنسی از سایر کروموزوم‌ها دو عدد دارند که یکی را از پدر و دیگری را از مادر به ارث 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 اسنپ (SNP)<sup>۱</sup> است. در جایگاه‌های اسنپ بیش از یک نوع باز بر روی ژنوم افراد مختلف جمعیت ملاحظه می‌شود که هر نوع متفاوت یک آلل نامیده می‌شود [۱-۳]. اگر یک جفت از آلل‌ها در یک مکان اسنپ مقادیر یکسان داشته باشند، مکان اسنپ هموزیگوت و در غیر این صورت هتروزیگوت است. مجموعه‌ی همه‌ی اسنپ‌های واقع بر روی یک کروموزوم که با هم به نسل بعد منتقل می‌شوند، هاپلوتا‌پ نامیده می‌شود. به عبارت دیگر هاپلوتا‌پ متشکل از نواحی ژنی نزدیک به هم است که باهم به نسل بعد منتقل می‌شوند. تقریباً تمام چندریختی‌ها از دو آلل متفاوت تشکیل شده‌اند که می‌توان آلل با بیشترین فراوانی در یک مکان اسنپ (آلل غالب) را با صفر و آلل دیگر (آلل مغلوب) را با یک کدگذاری نمود. با این کدگذاری یک هاپلوتا‌پ را می‌توان به صورت برداری از صفرها و یک‌ها نمایش داد [۴].

چون تعیین توالی هاپلوتا‌پ‌ها 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<sup>۲</sup> با روی هم قراردادن آن‌ها توالی کل ژنوم را می‌یابند. تفکیک قطعات توالی‌یابی شده به دو یا چند دسته و روی هم گذاری آن‌ها برای بازسازی هاپلوتا‌پ‌ها مساله‌ی برهم‌گذاری هاپلوتا‌پ نامیده می‌شود. اگر قطعات کوتاه توالی‌یابی شده فاقد خطا باشند به سادگی می‌توان برای گونه‌های دیپلوئید دو هاپلوتا‌پ مجهول را تعیین نمود. اما وجود خطا باعث چالش محاسباتی قابل توجهی می‌شود، به‌طوری که ثابت شده که مساله در این حالت در رده‌ی مسایل NP-hard قرار می‌گیرد [۵]. تحقیقات زیادی روی مساله‌ی برهم‌گذاری هاپلوتا‌پ در حالت دیپلوئید انجام شده است [۴، ۶-۱۲]. به‌طور کلی روش‌های مواجهه با مساله به دو دسته‌ی دقیق و تقریبی (ابتکاری) تقسیم‌بندی می‌شوند. روش‌های دقیق معمولاً مساله را به صورت یک مدل برنامه‌ریزی خطی دودویی فرمول‌بندی می‌کنند که برای حل آن‌ها می‌توان از حل‌کننده‌هایی نظیر گروبی<sup>۳</sup> یا سیپلکس<sup>۴</sup> بهره گرفت [۴، ۶]. این روش‌ها برای مسایل واقعی که حجم داده‌ها بسیار زیاد است به زمان اجرای بسیار طولانی نیاز دارند. در مقابل 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 هاپلوتا‌پ معرفی شده‌اند که از جمله می‌توان به کمترین تصحیح خطا (MEC)<sup>۵</sup> [۵]، حذف کمترین تعداد اسنپ (MSR)<sup>۱</sup>، حداقل حذف قطعه (MFR)<sup>۲</sup> [۱۳]، حداکثر

<sup>1</sup> Single Nucleotide Polymorphism

<sup>2</sup> Fragment

<sup>3</sup> Gurobi

<sup>4</sup> CPLEX

<sup>5</sup> Minimum Error Correction

برش قطعه (MFC) [۱۴] و افراز بهینه متعادل (BOP) [۱۵] اشاره کرد. در این مقاله روی تابع هدف کمترین تصحیح خطا متمرکز می‌شویم که تلاش می‌کند کمترین تعداد نوکلئوتیدها را در خوانش‌ها<sup>۵</sup> 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 در گونه‌های پلی‌پلوئید نتیجه می‌دهند اما مساله‌ی برهم‌گذاری هاپلوتایپ برای گونه‌های پلی‌پلوئید بسیار چالش برانگیزتر از گونه‌های دیپلوئید است. به همین دلیل تحقیقات بسیار کمی در این زمینه انجام شده است. الگوریتم HapCompass [۱۶] مساله‌ی تعیین هاپلوتایپ‌ها را به مساله‌ی یافتن یک درخت فراگیر از گراف ساخته شده توسط خوانش‌های DNA تبدیل می‌کند. الگوریتم HapTree [۱۷] که از برآورد گر حداکثر خطا استفاده می‌کند برای کاهش پیچیدگی فرآیند محاسبات از یک استراژی مشابه برنامه‌ریزی پویا استفاده می‌نماید. الگوریتم SDhap [۱۸] مساله را به صورت 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 در حالت دیپلوئید و هم در حالت تریپلوئید مورد استفاده قرار داد. الگوریتم H-POP [۱۹] مساله‌ی تعیین هاپلوتایپ‌ها را به صورت مساله‌ی دسته‌بندی خوانش‌ها مدل‌سازی می‌کند که (PBOP) نامیده می‌شود. برای خوانش‌های توالی‌یابی شده از یک ژنوم k-ploid مدل سعی می‌کند که خوانش‌ها را به k گروه دسته‌بندی کند، به طوری که تفاوت میان خوانش‌های هم‌دسته مینیمم و تفاوت میان خوانش‌های دسته‌های متفاوت ماکزیمم شود. وقتی اطلاعات مربوط به ژنوتایپ در دسترس باشد، مدل به مساله‌ی PBOPG<sup>۶</sup> 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 محدود کردن جواب‌های میانی در هر تکرار ارایه شده است.

در این 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. هر دو مدل مطرح شده در حالت وجود خطا در خوانش‌ها روی داده‌های شبیه‌سازی شده 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 را حل 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 ارایه شده توسط روش‌های ابتکاری و بهبود دقت جواب‌های آن‌ها می‌توان از روش‌های دقیق بهره گرفت.

<sup>1</sup> Minimum SNP Removal

<sup>2</sup> Minimum Fragment Removal

<sup>3</sup> Maximum Fragment Cut

<sup>4</sup> Balanced Optimal Partition

<sup>5</sup> reads

<sup>6</sup> Polyploid Balanced Optimal Partition

<sup>7</sup> Polyploid Balanced Optimal Partition with Genotype Constraints

## ۲ روش‌ها

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

$$d(r_i, r_j) = \sum_{k=1}^n d(r_{ik}, r_{jk}), \quad (1)$$

که در آن

$$d(r_{ik}, r_{jk}) = \begin{cases} 1 & r_{ik} \neq r_{jk} \neq - \\ 0 & \text{در غیر این صورت} \end{cases} \quad (2)$$

هدف مساله‌ی برهم‌گذاری هاپلوتا‌پ در حالت تریپلوئید تقسیم سطرهاى ماتریس  $M$  به سه دسته است به‌طوری که از هر دسته یک هاپلوتا‌پ نتیجه شود. هاپلوتا‌پ‌ها را به صورت (۳)، (۴) و (۵) نمایش می‌دهیم.

$$h_1 = \{h_1^1, h_1^2, \dots, h_1^n\}, \quad (3)$$

$$h_2 = \{h_2^1, h_2^2, \dots, h_2^n\}, \quad (4)$$

$$h_3 = \{h_3^1, h_3^2, \dots, h_3^n\}. \quad (5)$$

در مساله ی برهم گذاری هاپلوتایپ در حالت تریپلوئید تحت تابع  $MEC$  می خواهیم هاپلوتایپ های  $H = \{h_1, h_r, h_v\}$  را بیابیم به طوری که  $MEC(M, H)$  که با روابط (۶) و (۷) تعریف می شود مینیمم گردد.

$$MEC(M, H) = \sum_{i=1}^m MEC(r_i, H), \quad (6)$$

که در آن  $r_i$  سطر  $i$ ام ماتریس  $M$  و

$$MEC(r_i, H) = \min \{d(r_i, h_1), d(r_i, h_r), d(r_i, h_v)\}. \quad (7)$$

برای نوشتن مدل ریاضی مساله دو حالت زیر را در نظر می گیریم.

## ۱-۲ حالتی که اطلاعات ژنوتایپ در دسترس است

در این حالت تعداد یک های موجود در بین مولفه های نظیر هم در هاپلوتایپ ها را می دانیم که حداقل یک و حداکثر دو است، یعنی  $h_1^j = h_r^j = h_v^j = 1$  و  $h_1^j = h_r^j = h_v^j = 0$  رخ نمی دهند. در این حالت مدل های  $TILPG1$  و  $TILPG2$  را برای مساله ی برهم گذاری هاپلوتایپ ارایه می کنیم.

$TILPG1$ :

$$\text{Min } \sum_{i=1}^m \left( \sum_{j \in J_{i,0}} t_{ij} + \sum_{j \in J_{i,1}} u_{ij} \right) \quad (8)$$

s.t.

$$h_1^j + y_1^i - 1 \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,0} \quad (9)$$

$$h_v^j + y_v^i - 1 \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,0} \quad (10)$$

$$1 - (h_1^j + h_v^j) - y_1^i - y_v^i \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,0} \quad (11)$$

$$y_1^i - h_1^j \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (12)$$

$$y_v^i - h_v^j \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (13)$$

$$h_1^j + h_v^j - y_1^i - y_v^i \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (14)$$

$$y_1^i + y_v^i \leq 1, \quad \forall i = 1, \dots, m \quad (15)$$

$$h_1^j + h_v^j \leq 1, \quad \forall j = 1, \dots, n \quad (16)$$

$$y_1^i, y_v^i, h_1^j, h_v^j, t_{ij}, u_{ij} \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n. \quad (17)$$

$(J_{i,1})$  مجموعه‌ی ستون‌هایی در ماتریس  $M$  است که در سطر  $i$  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)  $y_i^i = 1$  و  $y_p^i = 0$  اگر و تنها اگر سطر  $i$  به هاپلوتا‌پ  $h_i$  تخصیص داده شود.

(ب)  $y_i^i = 0$  و  $y_p^i = 1$  اگر و تنها اگر سطر  $i$  به هاپلوتا‌پ  $h_p$  تخصیص داده شود.

(پ)  $y_i^i = 0$  و  $y_p^i = 0$  اگر و تنها اگر سطر  $i$  به هاپلوتا‌پ  $h_p$  تخصیص داده شود.

محدودیت‌های (۱۶) تضمین می‌کنند که حالت  $h_i^j = h_p^j = 1$  امکان وقوع ندارد.  $h_i^j$  و  $h_p^j$  به ترتیب نشان‌دهنده‌ی بیت  $z$  ام هاپلوتا‌پ‌های  $h_i$  و  $h_p$  هستند که می‌توانند صفر یا یک باشند. درحالی که اطلاعات ژنوتا‌پ را داشته باشیم به جای  $h_p^j$  می‌توانیم بنویسیم  $1 - (h_i^j + h_p^j)$ . متغیرهای باینری  $t_{ij}$  و  $u_{ij}$  متضاد بودن (جریمه) یا نبودن بیت  $z$  ام سطر  $i$  را با هاپلوتا‌پی که به آن تخصیص یافته است نشان می‌دهند و مقدار آن‌ها یک است اگر و فقط اگر بیت  $z$  ام سطر  $i$  با بیت  $z$  ام هاپلوتا‌پی که به آن تخصیص یافته در تضاد باشد. در غیر این صورت 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 هاپلوتا‌پ با آن هاپلوتا‌پ 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  $i \in \{1, \dots, m\}$  و هر  $j \in J_{i,0}$ ،  $t_{ij} = 1$  اگر و تنها اگر  $h_i^j = 1$ ،  $y_p^i = 0$ ،  $y_i^i = 1$  یا  $h_i^j = 1$ ،  $y_p^i = 1$ ،  $y_i^i = 0$  یا  $h_i^j = 1 - (h_i^j + h_p^j) = 1$ ،  $y_p^i = 0$ ،  $y_i^i = 0$ .

$h_i^j = 1$  و  $y_p^i = 0$ ،  $y_i^i = 1$  با هم به این معنا هستند که سطر  $i$  به هاپلوتا‌پ  $h_i$  تخصیص داده شده و درآیه‌ی  $z$  ام آن صفر است ( $j \in J_{i,0}$ ) و با درآیه‌ی  $z$  ام  $h_i$  متفاوت است ( $h_i^j = 1$ ).  $h_i^j = 1$ ،  $y_p^i = 1$ ،  $y_i^i = 0$  با هم به این معنا هستند که سطر  $i$  به هاپلوتا‌پ  $h_p$  تخصیص داده شده اما درآیه‌ی  $z$  ام آن با درآیه‌ی  $z$  ام  $h_p$  متفاوت است.  $h_i^j = 1 - (h_i^j + h_p^j) = 1$ ،  $y_p^i = 0$ ،  $y_i^i = 0$  به این معنا هستند که سطر  $i$  به هاپلوتا‌پ  $h_p$  تخصیص داده شده اما درآیه‌ی  $z$  ام آن با درآیه‌ی  $z$  ام  $h_p$  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 (۹)–(۱۱)،  $t_{ij}$  دارای سه کران پایینی  $1 - h_i^j + y_i^i$ ،  $1 - h_p^j + y_p^i$  و  $1 - (h_i^j + h_p^j) - y_i^i - y_p^i$  است. با تغییر مقادیر متغیرها این سه محدودیت کران‌های پایینی متفاوتی را برای  $t_{ij}$  ایجاد می‌کنند. برای مثال در حالتی که  $y_i^i = 1$  و  $y_p^i = 0$  داریم:

$$h_i^j + y_i^i - 1 = h_i^j \leq t_{ij} \quad (18)$$

$$h_p^j + y_p^i - 1 = h_p^j - 1 \leq t_{ij} \quad (19)$$

$$1 - (h_i^j + h_r^j) - y_i^i - y_r^i = -(h_i^j + h_r^j) \leq t_{ij}. \quad (20)$$

در بین سه کران پایینی به دست آمده از سه محدودیت (۱۸)، (۱۹) و (۲۰) برای متغیر  $h_i^j, t_{ij}$  از بقیه بزرگ‌تر است و لذا  $t_{ij}$  مساوی  $h_i^j$  خواهد شد. در نتیجه اگر  $h_i^j = 1, y_r^i = 0, y_i^i = 1$  آنگاه  $t_{ij} = 1$  است. یعنی در حالتی که  $j \in J_{i,0}$  (بیت  $j$  ام سطر  $i$  صفر است) و  $h_i^j = 1$  (بیت  $j$  ام هاپلوتا پ  $h_i$  یک است) و  $y_i^i = 1, y_r^i = 0$  (سطر  $i$  ام به هاپلوتا پ  $h_i$  تخصیص داده شده) از آن جایی که بیت‌های  $j$  ام سطر  $i$  ام و هاپلوتا پ  $h_i$  یکسان نیستند ولی این سطر به  $h_i$  تخصیص یافته، جریمه‌ای به اندازه‌ی یک واحد به تابع هدف اضافه می‌شود یعنی  $t_{ij} = 1$  می‌گردد. به طریق مشابه در حالتی که  $h_i^j = 1, y_r^i = 1, y_i^i = 0$  یا  $h_r^j = 1 - (h_i^j + h_r^j) = 1, y_i^i = 0, y_r^i = 0$  می‌توان نشان داد که  $t_{ij} = 1$  می‌شود. همچنین برای هر  $i \in \{1, \dots, m\}$  و هر  $j \in J_{i,1}$  اگر  $u_{ij} = 1$  و تنها اگر  $1 - h_i^j = 1, y_r^i = 0, y_i^i = 1$  یا  $1 - h_r^j = 1, y_i^i = 0, y_r^i = 1$  اثبات این مطلب مشابه حالت  $j \in J_{i,0}$  است با این تفاوت که از جدول ۲ استفاده می‌شود.

جدول ۱. مقادیر  $t_{ij}$  برای  $j \in J_{i,0}$ 

|          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $h_i^j$  | ۰ | ۱ | ۰ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ |
| $h_r^j$  | ۰ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۱ |
| $h_r^j$  | ۰ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۱ | ۱ |
| $y_i^i$  | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ |
| $y_r^i$  | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | 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 |

جدول ۲. مقادیر  $u_{ij}$  برای  $j \in J_{i,1}$ 

|          |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
|----------|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $h_i^j$  | ۰ | ۱ | ۰ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ |
| $h_r^j$  | ۰ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۱ |
| $h_r^j$  | ۰ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۱ | ۱ |
| $y_i^i$  | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ |
| $y_r^i$  | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۱ | ۰ | ۰ | ۰ | ۰ | ۰ | ۰ |
| $u_{ij}$ | ۱ | ۰ | ۱ | ۱ | ۰ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۱ | ۰ | ۱ | ۱ | ۰ | ۱ | ۰ | ۰ | ۰ |

مسئله‌ی برهم‌گذاری هاپلوتا پ 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 مسئله‌ی برهم‌گذاری هاپلوتا پ را در حالت تریپلوئید و با فرض در دسترس بودن داده‌های ژنوتایپ فرمول‌بندی می‌کند.

TILPG<sub>2</sub>:

$$\text{Min} \sum_{i=1}^m \left( \sum_{j \in J_{i,\circ}} \left[ 1 - (h_i^j + h_v^j) - (y_i^i + y_v^i) + t_{ij} + u_{ij} \right] + \sum_{j \in J_{i,\vee}} \left[ (h_i^j + h_v^j) - (y_i^i + y_v^i) + t_{ij} + u_{ij} \right] \right) \quad (21)$$

s.t.

$$2h_i^j + h_v^j + 2y_i^i - 2 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (22)$$

$$h_i^j + 2h_v^j + 2y_v^i - 2 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (23)$$

$$-2h_i^j - h_v^j + 2y_i^i \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\vee} \quad (24)$$

$$-h_i^j - 2h_v^j + 2y_v^i \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\vee} \quad (25)$$

$$y_i^i + y_v^i \leq 1, \quad \forall i = 1, \dots, m \quad (26)$$

$$h_i^j + h_v^j \leq 1, \quad \forall j = 1, \dots, n \quad (27)$$

$$y_i^i, y_v^i, h_i^j, h_v^j, t_{ij}, u_{ij} \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n. \quad (28)$$

متغیرهای مدل TILPG<sub>2</sub> مانند مدل TILPG<sub>1</sub> تعریف می‌شوند. محدودیت‌های (22)، (23)، (24) و (25) کران‌های پایینی مناسب برای  $t_{ij} + u_{ij}$  تولید می‌کنند. محدودیت‌های (26) نشان می‌دهند که حالت  $y_i^i = y_v^i = 1$  امکان وقوع ندارد. محدودیت‌های (27) تضمین می‌کنند که حالت  $h_i^j = h_v^j = 1$  امکان وقوع ندارد. همچنین درحالتی که اطلاعات ژنوتایپ را داشته باشیم به جای  $h_i^j$  می‌توانیم بنویسیم  $1 - (h_i^j + h_v^j)$ . اثبات این که مدل TILPG<sub>2</sub> نیز مانند مدل TILPG<sub>1</sub> به درستی سطرهای ماتریس ورودی را به سه دسته تقسیم می‌کند به طوری که سطرهای موجود در هر دسته با هاپلوتا‌پ نظیر آن کم‌ترین تضاد را داشته باشند به طریق مشابه با استفاده از جداول 3 و 4 انجام می‌شود.

برای هر  $i \in \{1, \dots, m\}$  و هر  $j \in J_{i,\circ}$   $1 - (h_i^j + h_v^j) - (y_i^i + y_v^i) + t_{ij} + u_{ij} = 1$ ، اگر و تنها اگر  $h_i^j = 1, y_i^i = 1, y_v^i = 0, h_v^j = 1$  یا  $y_i^i = 0, y_v^i = 1, h_i^j = 1$  یا  $h_i^j = 0, y_v^i = 0, h_v^j = 1$  یا  $y_i^i = 0, y_v^i = 0, h_v^j = 1$ . تعریف‌های این سه حالت مشابه تعریف‌های گفته شده برای آن‌ها در مدل TILPG<sub>1</sub> است. در جدول 3 تمامی مقادیر ممکن برای متغیرهای  $y_i^i, y_v^i, h_i^j, h_v^j$  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 محدودیت‌های مدل TILPG<sub>2</sub> در حالت  $j \in J_{i,\circ}$  دو کران پایین  $2h_i^j + h_v^j + 2y_i^i - 2$  و  $h_i^j + 2h_v^j + 2y_v^i - 2$  برای این عبارت وجود دارد. برای مثال در حالتی که  $y_i^i = 1, y_v^i = 0$  داریم:

$$2h_i^j + h_v^j + 2y_i^i - 2 = 2h_i^j + h_v^j \leq t_{ij} + u_{ij} \quad (29)$$

$$h_i^j + 2h_v^j + 2y_v^i - 2 = h_i^j + 2h_v^j - 2 \leq t_{ij} + u_{ij}. \quad (30)$$



TILPNG۱:

$$\text{Min} \sum_{i=1}^m \left( \sum_{j \in J_{i,\circ}} t_{ij} + \sum_{j \in J_{i,1}} u_{ij} \right) \quad (31)$$

s.t.

$$h_{\gamma}^j + y_{\gamma}^i - 1 \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (32)$$

$$h_{\gamma}^j + y_{\gamma}^i - 1 \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (33)$$

$$h_{\gamma}^j - y_{\gamma}^i - y_{\gamma}^i \leq t_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (34)$$

$$y_{\gamma}^i - h_{\gamma}^j \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (35)$$

$$y_{\gamma}^i - h_{\gamma}^j \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (36)$$

$$1 - (y_{\gamma}^i + y_{\gamma}^i) - h_{\gamma}^j \leq u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (37)$$

$$y_{\gamma}^i + y_{\gamma}^i \leq 1, \quad \forall i = 1, \dots, m \quad (38)$$

$$y_{\gamma}^i, y_{\gamma}^i, h_{\gamma}^j, h_{\gamma}^j, t_{ij}, u_{ij} \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n \quad (39)$$

و

TILPNG۲:

$$\text{Min} \sum_{i=1}^m \left( \sum_{j \in J_{i,\circ}} [h_{\gamma}^j - (y_{\gamma}^i + y_{\gamma}^i) + t_{ij} + u_{ij}] + \sum_{j \in J_{i,1}} [1 - h_{\gamma}^j - (y_{\gamma}^i + y_{\gamma}^i) + t_{ij} + u_{ij}] \right) \quad (40)$$

s.t.

$$h_{\gamma}^j - h_{\gamma}^j + 2y_{\gamma}^i - 1 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (41)$$

$$h_{\gamma}^j - h_{\gamma}^j + 2y_{\gamma}^i - 1 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,\circ} \quad (42)$$

$$-h_{\gamma}^j + h_{\gamma}^j + 2y_{\gamma}^i - 1 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (43)$$

$$-h_{\gamma}^j + h_{\gamma}^j + 2y_{\gamma}^i - 1 \leq t_{ij} + u_{ij}, \quad \forall i = 1, \dots, m, \forall j \in J_{i,1} \quad (44)$$

$$y_{\gamma}^i + y_{\gamma}^i \leq 1, \quad \forall i = 1, \dots, m \quad (45)$$

$$y_{\gamma}^i, y_{\gamma}^i, h_{\gamma}^j, h_{\gamma}^j, h_{\gamma}^j, t_{ij}, u_{ij} \in \{0, 1\}, \quad \forall i = 1, \dots, m, \forall j = 1, \dots, n. \quad (46)$$

تعاریف متغیرهای مدل‌های  $TILPNG_1$  و  $TILPNG_2$  همانند مدل‌های  $TILPG_1$  و  $TILPG_2$  است. یک متغیر جدید  $h_p^j$  در مدل‌های  $TILPNG_1$  و  $TILPNG_2$  وجود دارد که جایگزین  $1 - (h_p^j + h_p^j)$  در مدل‌های  $TILPG_1$  و  $TILPG_2$  شده است. همچنین اثبات درستی مدل‌های  $TILPNG_1$  و  $TILPNG_2$  مشابه مدل‌های  $TILPG_1$  و  $TILPG_2$  با استفاده از جداول ۱، ۲، ۳ و ۴ انجام می‌شود با این تفاوت که در این مدل‌ها حالت‌های  $h_1^j = h_p^j = h_p^j = 0$  و  $h_1^j = h_p^j = h_p^j = 1$  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 هاپلوتا پ در حالت تریپلوئید طراحی شده‌اند، مدل‌های ارائه‌شده 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 ۴۷۹۰-iv مورد ارزیابی قرار دادیم. در شبیه‌سازی داده‌ها سه پارامتر لحاظ شدند. طول هاپلوتا پ، نرخ خطا و نرخ پوشش که به ترتیب با  $e$ ،  $l$  و  $c$  نمایش داده می‌شوند. ما با مقادیر  $e = 0/1$ ،  $l = 10, 20, 30, 40$  و  $c = 3, 4, 5$  داده تولید نمودیم. نرخ پوشش بیانگر این است که حداکثر تعداد صفرها و یک‌ها در هر ستون ماتریس ورودی  $3c$  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  $TILPG_1$  و  $TILPG_2$  توسط حل‌کننده‌ی AIMMS آورده شده است.

**جدول ۵.** متوسط زمان اجرا و MEC روی ۱۰ ماتریس با  $l = 10, 20, 30, 40$  و  $c = 3, 4, 5$  و  $e = 0/1$  در حالت در دسترس بودن اطلاعات

ژنوتایپ

| (1,c)  | مدل $TILPG_1$ |                 | مدل $TILPG_2$ |                 |
|--------|---------------|-----------------|---------------|-----------------|
|        | MEC           | زمان حل (ثانیه) | MEC           | زمان حل (ثانیه) |
| (10,3) | ۱/۹           | ۰/۰۲            | ۱/۹           | ۰/۰۷            |
| (10,4) | ۳/۳           | ۰/۰۴۶           | ۳/۳           | ۰/۱۵۲           |
| (10,5) | ۵/۲           | ۰/۰۸            | ۵/۲           | ۰/۳۰۶           |
| (20,3) | ۱۴/۱          | ۰/۲۸۷           | ۱۴/۱          | ۴/۵۱۲           |
| (20,4) | ۱۸/۲          | ۱/۰۲۶           | ۱۸/۲          | ۸/۴۲۲           |
| (20,5) | ۲۵/۷          | ۱/۳۶۴           | ۲۵/۷          | ۳۱/۲۵۵          |
| (30,3) | ۲۴/۴          | ۲/۴۱            | ۲۴/۴          | ۲۰/۱۰۱          |
| (30,4) | ۳۱/۳          | ۷/۵۶۸           | ۳۱/۳          | ۵۲/۴۹۱          |
| (30,5) | ۴۲/۲          | ۲۸/۹۳۶          | ۴۲/۲          | ۲۴۸/۸۰۹         |
| (40,3) | ۳۲/۸          | ۳/۱۲۳           | ۳۲/۸          | ۲۵/۴۵۵          |
| (40,4) | ۴۷/۶          | ۳۱/۳۷۸          | ۴۷/۶          | ۱۶۳/۴۲۲         |
| (40,5) | ۵۷/۹          | ۲۷۲/۰۶۹         | ۵۷/۹          | ۲۹۷۶/۸۴۳        |

همان‌طور که ملاحظه می‌شود مدل  $TILPG_1$  نسبت به مدل  $TILPG_2$  در زمان کوتاه‌تری به جواب می‌رسد. علاوه بر این با افزایش  $l$  و  $c$  زمان حل مدل‌ها افزایش می‌یابد و تفاوت زمان حل دو مدل بیشتر نمایان می‌گردد. در جدول ۶ مدل‌های  $TILPNG_1$  و  $TILPNG_2$  که مربوط به حالتی هستند که اطلاعات ژنوتا‌پ در دسترس نیست از نظر زمان حل با هم مورد مقایسه قرار گرفته‌اند. سیستم برای حل ۱۰ ماتریس با طول و نرخ پوشش  $l = 40$  و  $c = 5$  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  $TILPNG_2$  نیز برابر با مقدار MEC مدل  $TILPNG_1$  یعنی  $51/9$  باشد.

**جدول ۶.** متوسط زمان اجرا و MEC روی ۱۰ ماتریس با  $l = 10, 20, 30, 40$  و  $c = 3, 4, 5$  و  $e = 0/1$  در حالت عدم دسترسی به اطلاعات ژنوتا‌پ

| $(l, c)$ | مدل $TILPNG_1$ |                 | مدل $TILPNG_2$ |                 |
|----------|----------------|-----------------|----------------|-----------------|
|          | MEC            | زمان حل (ثانیه) | MEC            | زمان حل (ثانیه) |
| (10, 3)  | 0/7            | 0/075           | 0/7            | 0/099           |
| (10, 4)  | 1/9            | 0/136           | 1/9            | 0/167           |
| (10, 5)  | 3/9            | 0/268           | 3/9            | 0/564           |
| (20, 3)  | 10/3           | 2/988           | 10/3           | 4/723           |
| (20, 4)  | 14/2           | 8/496           | 14/2           | 12/233          |
| (20, 5)  | 20/3           | 35/23           | 20/3           | 111/946         |
| (30, 3)  | 18/3           | 12/489          | 18/3           | 25/043          |
| (30, 4)  | 27/9           | 114/528         | 27/9           | 378/057         |
| (30, 5)  | 35/8           | 701/566         | 35/8           | 4269/792        |
| (40, 3)  | 24/8           | 32/514          | 24/8           | 89/851          |
| (40, 4)  | 36/3           | 212/29          | 36/3           | 629/239         |
| (40, 5)  | 51/9           | 9028/164        | 51/9           | -               |

#### ۴ بحث

در این مقاله دو مدل برنامه‌ریزی خطی دودویی برای مساله‌ی برهم‌گذاری هاپلوتا‌پ در حالت تریپلوئید و با فرض وجود و عدم وجود اطلاعات ژنوتا‌پ طراحی شدند. عملکرد مدل‌ها از نظر زمان اجرا روی داده‌های شبیه‌سازی شده 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 هاپلوتا‌پ 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.