|  |
| --- |
| **3)شرح موضوع پیشنهادی: (تعریف مساله، فرضیات و هدف ...):** (هنگام تایپ فضای لازم باز خواهد شد مطالب به گونه­ای تایپ شود که در همین صفحه جا شود. توضیحات مفصل در پروپوزال تشریحی مجزا ارائه گردد. فایل مربوطه در سایت دانشگاه موجود است.)  **مقدمه**  تجزیه چندضلعی ساده[[1]](#footnote-1) فرآیندی است که در آن یک چندضلعی ساده به مجموعه­ای از چندضلعی­های کوچکتر افراز می­شود. در بعضی مواقع لازم است افراز به نحوی انجام شود که تعداد چندضلعی­های تولید­­ شده کمینه باشد]1[. همچنین ممکن است لازم باشد عمل تجزیه را به گونه­ای انجام دهیم که چندضلعی­های حاصل از آن محدب[[2]](#footnote-2) و­ یا یکنوا[[3]](#footnote-3) باشند. در مواردی نیز هدف صرفا تجزیه چندضلعی به مجموعه­ای از چندضلعی­های ساده­تر و کوچکتر است و الزامی وجود ندارد که چندضلعی­های حاصل از افراز ویژگی خاصی داشته باشند. تجزیه چندضلعی­ها در حوزه­های مختلفی مانند گرافیک برداری، تشخیص الگو، تشخیص متن، محاسبه جمع­های مینکوفسکی و طرح­ریزی حرکت ربات[[4]](#footnote-4) کاربرد دارد]1[. از آنجایی که اعمال کردن اکثر الگوریتم­ها روی چندضلعی­های محدب یا یکنوا، ساده­تر و کم­هزینه­تر از چندضلعی­های ساده­ای است که این ویژگی­ها را ندارند، تمایل زیادی برای یافتن الگوریتم­های کارا جهت تجزیه چندضلعی­های ساده وجود دارد]1،2[.  تا به حال الگوریتم­های مختلفی برای تجزیه چندضلعی­ها ارائه شده است. اکثر آن­ها برای چندضلعی­های ساده طراحی شده­اند ولی الگوریتم­هایی نیز وجود دارند که برای چندضلعی­های ساده و غیر ساده قابل استفاده­اند. تا جایی که اطلاع داریم تا کنون هیچ الگوریتم موازی برای حل مسئله تجزیه چندضلعی ارائه نشده است.  **تعریف مساله**  هدف ارائه یک الگوریتم (کارا) برای حل مسئله «تجزیه یک چندضلعی ساده به مجموعه­ای از چندضلعی­های یکنوا با تعداد کمینه» است. در این مسئله باید یک چندضلعی را به گونه­ای تجزیه کنیم که تعداد  چندضلعی­های حاصل از آن، از هر تجزیه دیگری کمتر باشد (تعداد چندضلعی­های حاصل کمینه باشد). همچنین همه چندضلعی­های تولید شده می­بایست یکنوا باشند. یکنوا بودن یک چندضلعی نسبت به یک خط تعریف  می­شود. یک چندضلعی را نسبت به خط L یکنوا گوییم اگر و تنها اگر هر خط عمود بر L یا سطح چندضلعی را اصلا قطع نکند و یا فقط یک بار سطح چندضلعی را قطع کند (به عبارت دیگر اشتراک خط L با چندضلعی یک پاره­خط پیوسته باشد). چنانچه قادر باشیم خطی عمود بر L رسم کنیم که سطح چندضلعی را دو بار یا بیشتر قطع کند، چندضلعی مفروض نسبت به خط L یکنوا نیست. تعریف رسمی مسئله به صورت زیر است:  *ورودی الگوریتم:* یک ساختار لیست پیوندی که اطلاعات اضلاع و رئوس چندضلعی (ساده) را نگهداری می­کند.  *خروجی الگوریتم:* مجموعه ای از چندضلعی­های یکنوا، به­طوری که اجتماع آن­ها برابر چندضلعی ورودی مسئله می­باشد و هیچ اشتراکی با هم ندارند (یک افراز از چندضلعی ورودی). این چندضلعی­ها با اضافه کردن قطرهایی به چندضلعی اولیه بوجود می­آیند.  **پیشینه:**  در سال 1983 *جی. ام. کیل*[[5]](#footnote-5) نشان داد که مسئله «تجزیه یک چندضلعی *حفره­دار* به چندضلعی­های یکنوا با تعداد کمینه» یک مسئله چندجمله­ای غیرقطعی کامل[[6]](#footnote-6) است]3[. همچنین *کیل* برای تجزیه یک چندضلعی­ ساده به صورت کمینه یک الگوریتم از مرتبه O(N n4) ارائه کرد]3[. در اینجا n تعداد کل رئوس چندضلعی و N تعداد راس­هایی از چندضلعی است که زاویه آن­ها از 180 درجه بیشتر است. برای حالتی که نیاز نیست تجزیه به صورت کمینه انجام شود (تعداد چندضلعی­های حاصل از تجزیه مهم نیست) *گری[[7]](#footnote-7)* یک الگوریتم از مرتبه O(n log n) ارائه کرد]4[. اگر نیاز باشد همه چندضلعی­های حاصل شده از تجزیه نسبت به یک خط خاص یکنواخت باشند، مسئله را «تجزیه یک چندضلعی ساده به مجموعه ای از چندضلعی­های یکپارچه یکنواخت» می­نامند. در سال 1988 *لیو و نتافوس[[8]](#footnote-8)* یک الگوریتم از مرتبه O(n N3 + N2 n log n + N5) برای حل این مسئله ارائه کردند]2[. الگویتم *لیو و نتافوس* در حالت کلی کاراتر از الگوریتم *کیل* است زیرا در عمل مقدار N از مقدار n کوچکتر است. برای یافتن الگوریتمی بهتر از الگوریتم *لیو و نتافوس* تحقیقات زیادی انجام دادیم ولی به نتیجه­ای نرسیدیم. به نظر می­رسد که الگوریتم *لیو و نتافوس* بهترین الگوریتم موجود برای حل مسئله «تجزیه یک چندضلعی ساده به چندضلعی­های یکپارچه یکنواخت کمینه» باشد. مقدار N (تعداد رئوس با زاویه بزرگتر از 180 درجه) معمولا عدد ثابتی نیست. با وجود اینکهN<n است ولی در بدترین حالت است، لذا الگوریتم *لیو و نتافوس* در بدترین حالتاز مرتبه O(n5) خواهد بود. با توجه به اینکه در کاربردهای عملی تعداد راس­ها (n) معمولا زیاد است، الگوریتم *لیو و نتافوس* در بدترین حالت زمان اجرای نسبتا زیادی خواهد داشت. این موضوع مخصوصا در پردازش­های بلادرنگ (مانند بازی­های گرافیکی سه بعدی) مشکل­ساز خواهد شد.  **هدف**  دستیابی به یک الگوریتم کاراتر جهت حل مسئله «تجزیه یک چندضلعی ساده به مجموعه­ای از چندضلعی­های یکنوا با تعداد کمینه». به سه روش می­توان به این هدف دست یافت:   1. موازی سازی بهترین الگوریتم سریال موجود   تا جایی که اطلاع داریم هنوز چنین کاری صورت نگرفته است.   1. بهبود یکی از الگوریتم های موجود   تا این لحظه الگوریتم ارائه شده توسط *لیو و نتافوس* بهترین گزینه به شمار می­رود زیرا (با وجود تحقیقات به عمل آمده) از الگوریتم کاراتری مطلع نیستیم. *لیو و نتافوس* در الگوریتم پیشنهادی خود برای حل مسئله تجزیه چندضلعی مجبور به حل یک مسئله از نظریه گراف شدند]2[. الگوریتم­ای که آن­ها برای حل آن مسئله (از نظریه گراف) استفاده کرده بودند به مراتب ناکارآمدتر از الگوریتم­هایی است که در سال­های بعد توسط دیگران ارائه شد]5[. اولین بهبودی که می­توان روی الگوریتم *لیو و نتافوس* داد این است که از الگوریتم­های کاراتر برای حل مسئله ذکر شده (از نظریه گراف) استفاده کرد.   1. ارائه یک الگوریتم سریال یا موازی جدید برای حل مسئله تجزیه چندضلعی   **مراجع**  [1] J.M. Keil, Polygon Decomposition, University of Saskatchewan, Saskatoon Sask., Canada. (1996) S7N 5A9  [2] R. Liu, S. Ntafos, On decomposing polygons into uniformly monotone components. Information Processing Letters (1988); 27:85-9.  [3] J.M. Keil, Decomposing a polygon into simpler components. Ph.D. thesis, Univ. of Toronto. Toronto. Canada. 1983. Report 163/83  [4] M.R. Garey, D.S. Johnson, F.P. Preparata, R.E. Tarjan, Triangulating a simple polygon. Information Processing Letters (1978) 7:175-179  [5] N. Nash, D. Gregg, An output sensitive algorithm for computing a maximum independent set of a circle graph, Information Processing Letters 110 (2010) 630-634 |
| امضای دانشجو امضای استاد راهنمای اول امضای استاد راهنمای دوم امضای استاد مشاور اول امضای استاد مشاور دوم  تاریخ |

دانشجوی گرامی ضمن آرزوی موفقیت برای شما به خاطر بسپارید. صدور مجوز دفاع در مدیریت تحصیلات تکمیلی دانشگاه حداکثر تا یک هفته مانده به دفاع انجام می­پذیرد در غیر اینصورت مجوز دفاع صادر نخواهد شد.

1. Decomposition of Simple Polygon [↑](#footnote-ref-1)
2. Convex [↑](#footnote-ref-2)
3. Monotone [↑](#footnote-ref-3)
4. Robot Motion Planning [↑](#footnote-ref-4)
5. J.M. Keil [↑](#footnote-ref-5)
6. NP-Complete [↑](#footnote-ref-6)
7. Garey [↑](#footnote-ref-7)
8. Liu and Ntafos [↑](#footnote-ref-8)