



مرکز توسعه فناوری نیرو (متن)



انجمن کامپیوتر ایران
Computer Society of Iran

زمان بندی بار محاسباتی تقسیم پذیر با در نظر گرفتن زمان بازگشت نتایج در سیستم های ناهمگن با استفاده از الگوریتم PSO

<p>زهرا جوادی، فارغ التحصیل کارشناسی کامپیوتر، موسسه آموزش عالی سجاد، گروه کامپیوتر، Zahra_javadi42@yahoo.com</p>	<p>جواد حمیدزاده، دانشجوی دکتری کامپیوتر، دانشگاه فردوسی مشهد، دانشکده مهندسی، گروه کامپیوتر عضو هیئت علمی موسسه آموزش عالی سجاد Ja_ha47@stu-mail.um.ac.ir</p>	<p>رضا منصفی، عضو هیئت علمی دانشگاه فردوسی، دانشگاه فردوسی مشهد، دانشکده مهندسی، گروه کامپیوتر monsefi@um.ac.ir</p>
-------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------

جمله مسائل سخت است. یکی از مدل های محاسباتی مطرح در این سیستم ها، مدل بار محاسباتی تقسیم پذیر است. تئوری این دسته از محاسبات برای اولین بار در سال ۱۹۸۸ میلادی توسط آقای «برتازی» در مقاله [۱۳] معرفی شد که با چاپ کتاب [۲] توسط ایشان و همکاران، باعث توجه بیشتر محققین به این تئوری شد.

در تئوری بار محاسباتی تقسیم پذیر، بار محاسباتی از تعداد زیادی واحدهای محاسباتی ریزدانه تشکیل شده است که پردازش این واحدها می تواند به طور کاملاً مستقل و موازی با هم انجام گیرد. فرض بر این است که این واحدهای محاسباتی قابل تقسیم نباشند. در نتیجه، حجم زیادی از محاسبات، می تواند به هر نسبتی بر حسب واحدهای محاسباتی ریزدانه تقسیم شده و سپس برای اجرای موازی به ماشین های موجود ارسال شود.

این تئوری ابزار مناسبی جهت مدل سازی کاربردهای بسیاری با داده های حجیم است [۹-۷]. از جمله کاربردهایی که در این دسته از محاسبات می توان به آنها اشاره نمود عبارتند از: پردازش تصویر، پردازش سیگنال، جستجو در بانک های اطلاعاتی، داده کاوی، محاسبات جبرخطی و کاربردهای چند رسانه ای.

در این تئوری، از مدل ارباب و کارگر جهت پیاده سازی استفاده می شود. کل بار محاسباتی در این تئوری بر روی کامپیوتر ارباب قرار دارد که از طریق یک شبکه ارتباطی دارای همبندی مشخص، با کامپیوترهای کارگر در ارتباط است. کامپیوتر ارباب موظف است بار محاسباتی تقسیم پذیر را بین کامپیوترهای کارگر تقسیم نماید و کامپیوترهای کارگر تا دریافت کامل سهم بار محاسباتی خود منتظر مانده و سپس پردازش خود را آغاز نمایند. هر کامپیوتر کارگر پس از اتمام پردازش، موظف است طی ترتیبی که در زمان بندی مشخص شده نتیجه را به

چکیده: امروزه مسئله زمان بندی کارها در سیستم های ناهمگن به دلیل لزوم استفاده بهینه از ماشین های محاسباتی موجود و همچنین صرف زمان کمتر برای اجرای الگوریتم های زمان بندی، از اهمیت خاصی برخوردار است. در این مقاله زمان بندی بار محاسباتی تقسیم پذیر با در نظر گرفتن زمان بازگشت نتایج در یک سیستم ناهمگن دارای شبکه ارتباطی درختی تک سطحی بررسی شده است. یکی از اهداف زمان بندی در این گونه سیستم ها، کمینه سازی زمان کل پاسخ است. تاکنون الگوریتمی معین با پیچیدگی زمانی چند جمله ای که بتواند در تمام حالت ها جواب بهینه را تولید کند، برای این منظور ارائه نشده است. این مسئله مانند مسائل ترکیباتی، پیچیده به نظر می رسد و راه حل های موجود برای آن، راه حل های ابتکاری است. در این مقاله الگوریتم PSO به عنوان یک راه حل مسئله پیشنهاد شده است. با انجام شبیه سازی و مقایسه نتایج مشاهده می شود که این راه حل، در مقایسه با سایر روش های موجود جواب های بهتری تولید می کند. در میان روش های موجود، الگوریتم PSO پیشنهادی دارای کمترین میانگین درصد خطای نسبی است.

واژه های کلیدی: زمان بندی، بار محاسباتی تقسیم پذیر، زمان بازگشت نتایج، سیستم های ناهمگن، الگوریتم PSO.

- مقدمه

امروزه کارهای محاسباتی حجیم و زمان بندی اجرای آنها در سیستم های ناهمگن بسیار مورد توجه قرار گرفته است. در این گونه سیستم ها، کارایی ماشین ها، همبندی شبکه ارتباطی و سرعت خطوط ارتباطی شبکه می تواند متفاوت باشد بطوری که یک سیستم ناهمگن تشکیل شود. مسئله تطبیق و زمان بندی کارها در این سیستم ها، از

کامپیوتر ارباب بازگرداند. هدف از زمان بندی در این سیستم، تعیین سهم هر کامپیوتر کارگر، ترتیب توزیع سهمها و ترتیب بازگشت نتایج پردازش شده به کامپیوتر ارباب است، به گونه ای که زمان کل پاسخ کمینه شود.

امروزه تحقیقات در این زمینه با در نظر گرفتن تاخیر زمانی برای بازگشت نتیجه پردازشها مطرح گردیده است، به این ترتیب که اگر m ، تعداد کامپیوترهای کارگر باشد، با در نظر گرفتن ترتیبهای مختلف جهت توزیع و بازگشت نتیجه، پیچیدگی زمانی راه حل بهینه، از مرتبه $O(m^2)$ خواهد شد و چنانچه تعداد کامپیوترهای کارگر زیاد باشد، بدست آوردن زمان بندی بهینه، بسیار زمانبر خواهد بود. تاکنون نیز راه حلی بهینه با پیچیدگی زمانی چند جمله ای برای این مسئله ارائه نشده است. اما روشهای ابتکاری موجود برای حل این مسئله عبارتند از: روشهای LIFO، FIFO [۱۶ و ۱۴]، ITERLP [۱۵] و SPORT [۱۰]، ژنتیک [۱].

یکی از روشهای ابتکاری هوشمند در حل این مسئله، استفاده از روشهای محاسبات تکاملی، از جمله الگوریتم PSO است. تنها الگوریتم PSO ارائه شده در این زمینه الگوریتم [۱۲] است که در آن، تاخیر زمانی بازگشت نتیجه در نظر گرفته نشده است. هدف ما در این مقاله، پیشنهاد یک الگوریتم PSO برای حل این مسئله است که در آن، تاخیر زمانی بازگشت نتیجه در نظر گرفته شود.

ادامه این مقاله از بخشهای زیر تشکیل شده است. در بخش دوم مسئله زمان بندی بار تقسیم پذیر در یک شبکه دارای همبندی درختی تک سطحی بیان شده است. در بخش سوم، الگوریتم PSO جهت این زمان بندی بیان شده است. در بخش چهارم، نتایج شبیه سازیهای انجام شده و مقایسهها بیان گردیده است و در نهایت، در بخش پایانی نتیجه گیری و پیشنهاد ادامه کار بیان شده است.

- بیان مدل مسئله زمان بندی بار تقسیم پذیر

در این مدل، کامپیوتر ارباب در ریشه و کامپیوترهای کارگر در برگها واقع شده اند. در این مدل، E_k بیانگر مدت زمان لازم جهت پردازش یک واحد محاسباتی بر روی کامپیوتر کارگر شماره k است؛ C_k بیانگر مدت زمان لازم جهت ارسال یک واحد داده محاسباتی از کامپیوتر ارباب به کامپیوتر کارگر شماره k است. در این مدل، L بیانگر کل حجم بار محاسباتی اولیه موجود در کامپیوتر ارباب است. در این مدل $\alpha = \{\alpha_1, \dots, \alpha_m\}$ بیانگر درصد سهم بار محاسباتی هر یک از کامپیوترهای کارگر است؛ پارامتر m بیانگر تعداد کامپیوترهای کارگر است؛ پارامتر δ نیز بیانگر نسبت حجم داده حاصل از پردازش در هر کامپیوتر کارگر به سهم بار محاسباتی آن است.

- الگوریتم PSO برای بهینه سازی زمان بندی بار محاسباتی تقسیم پذیر

الگوریتمهای PSO، الگوریتمهای جستجوی مبتنی بر جمعیت هستند. برای حل یک مسئله، این الگوریتمها، با یک مجموعه از راه حل های تصادفی آغاز می گردند که دسته اولیه نامیده می شود. هر عضو این دسته، یک ذره نامیده می شود [۴]. هر ذره نشان دهنده یک راه حل مورد بررسی است و شامل دو بخش است. یک بخش، بیانگر ترتیب σ_a (ترتیب ارسال) و بخش دیگر بیانگر ترتیب σ_c (ترتیب بازگشت نتایج) است. تعداد عناصر هر بخش برابر m است که در این مسئله، برابر است با تعداد کامپیوترهای کارگر.

همانطور که قبلا هم گفته شد، در اینجا به دنبال یافتن ترتیبهای بهینه σ_a و σ_c هستیم، در نتیجه مسئله مورد بررسی ماهیت گسسته دارد و بنابراین از نسخه دودویی الگوریتم PSO برای حل نمودن این مساله استفاده می نماییم. همانطور که می دانیم اولین مرحله در پیاده سازی الگوریتم های جستجوی تکاملی فرموله کردن مساله است، یعنی یافتن یک روش کدگذاری مناسب، که در صورت یافتن یک روش کدگذاری مناسب مساله به راحتی قابل حل است. به عبارت دیگر ذره ها باید به گونه ای کدگذاری شوند که بتوانند نشان دهنده σ_a و σ_c باشند و در اینجا چون هر ذره نیاز به یک سرعت نیز دارد پس همان روش کدگذاری ذره ها را برای کدگذاری بردار سرعت نیز استفاده می نماییم.

۳-۱- موقعیت (مکان) ذره ها

یکی از موضوعات کلیدی در طراحی یک الگوریتم PSO مناسب، مرحله بازنمایی دانش (فرموله کردن مسئله) است که با هدف یافتن یک انطباق مناسب بین راه حل مسئله و ذره در PSO انجام می گیرد. در PSO پیشنهادی، هر ذره به صورت یک ماتریس $m \times m$ کدگذاری می شود که ماتریس موقعیت نامیده می شود. در اینجا ستون های ماتریس $m \times m$ موقعیت، همان شماره کامپیوترهای کارگر است و هر سطر از آن ماتریس نشان دهنده یک نوبت در ترتیب ارسال و دریافت است (ذره ها دوتوانند که یک نوع از آنها نشان دهنده σ_a و نوع دیگر آنها نشان دهنده σ_c است، به عبارتی دسته ما دو قسمتی است، که البته هر یک از ذره های یک بخش، متناظر است با یک ذره در بخش دیگر).

ماتریس موقعیت هر ذره دارای سه ویژگی زیر است:

- همه عناصر این ماتریس از مقادیر صفر یا یک تشکیل شده اند.
- در هر ستون از این ماتریس ها فقط یک عنصر ۱ است و بقیه عناصر صفر هستند.
- در هر سطر از این ماتریس ها فقط یک عنصر ۱ است و بقیه عناصر صفر هستند.

در ماتریس موقعیت هر ستون نمایش دهنده یک کامپیوتر کارگر است و هر سطر نمایش دهنده یک نوبت ارسال یا دریافت است، در هر ستون از یک ماتریس موقعیت مشخص می شود که یک کامپیوتر کارگر)

$$V_i(t+1)(j,k) = w * V_i(t)(j,k) + C1 * r1 * (pbest_i(t)(j,k) - X_i(t)(j,k)) + C2 * r2 * (nbest_i(t)(j,k) - X_i(t)(j,k)) \quad (1)$$

$$\begin{aligned} \text{If } V_i(t+1)(j,k) &= \max \{ V_i(t+1)(j,k) \} \\ \text{Then } X_i(t+1)(j) &= 1 \\ \text{Else } X_i(t+1)(j,k) &= 0 \end{aligned} \quad (2)$$

در معادله‌های (۱) و (۲)، $V_i(t+1)(j,k)$ یعنی عنصر سطر k ام و ستون k ام از آمین ماتریس سرعت که متناظر است با آمین ماتریس موقعیت (آمین ذره) در آمین مرحله زمانی الگوریتم مذکور و $X_i(t+1)(j,k)$ نشان دهنده عنصر سطر k ام و ستون k ام از آمین ماتریس موقعیت (آمین ذره) در آمین مرحله زمانی الگوریتم مذکور است.

۳-۴ ارزیابی نمودن برازندگی یک ذره

در اینجا زمان کل پاسخ (زمان ارسال بار به اولین کامپیوتر کارگر و دریافت همه نتایج از همه کامپیوترهای کارگر) به عنوان مقدار برازندگی در نظر گرفته شده است که چون هدف این است که مجموعه‌های σ_a ، σ_c و α را به گونه‌ای بدست آوریم که این زمان کمینه شود، پس هر ذره که مقدار برازندگی کمتری (min T) دارد، ذره بهتری است و پاسخ بهتری را نشان می‌دهد. تابع برازندگی، مشابه مقاله [۱] در نظر گرفته شده است.

- نتایج شبیه‌سازی و مقایسه‌ها

در آزمایش‌های انجام شده، کارایی الگوریتم PSO پیشنهادی را به کمک الگوریتم بهینه، با چهار الگوریتم ITERLP.LIFOC، SPORT و ژنتیک مقایسه کرده‌ایم. پیاده‌سازی‌ها، در محیط MatLab و بر روی کامپیوتری با پردازنده Intel Core 2 Due 1.73 GHz و دارای 3GB حافظه اصلی انجام شده است. مقادیر پارامترهای C و E جهت انجام آزمایش‌ها، برای نمایش یک سیستم ناهمگن، مشابه روش ارائه شده در مقاله [۱۰] تولید شده‌اند که در آن ۲۵ نمونه آزمایشی مختلف در نظر گرفته شده است. برای هر یک از این ۲۵ نمونه، مقدار برای پارامترهای C و E، در محدوده‌های تعیین شده، بصورت تصادفی و با توزیع یکنواخت تولید شده‌اند. مقادیر پارامترهای الگوریتم PSO در جدول شماره (۱) نشان داده شده است.

جدول(۱): مقادیر پارامترهای الگوریتم PSO

نام پارامتر	وظیفه	مقدار
C1	ضریب تجربه فردی	1.0
C2	ضریب تجربه اجتماعی	3.0
maxit	ماکزیمم تعداد نسل‌ها	100
N	اندازه جمعیت	10

کامپیوتر کارگر متناظر با اندیس آن ستون (در کدام نوبت از ترتیب ارسال یا دریافت باید قرارگیرد.

شکل(۱) نشان می‌دهد (با فرض اینکه این ذره نشان‌دهنده ترتیب دریافت سهم بار محاسباتی باشد) که کامپیوتر کارگر شماره ۱ (P1) دومین فردی است که سهم بار خود را دریافت می‌کند و کامپیوتر کارگر شماره ۲ (P2) نفر چهارم و کامپیوتر کارگر شماره ۳ (P3) نفر اول و کامپیوتر کارگر شماره ۴ (P4) نفر سومی است که سهم بار محاسباتی خود را دریافت می‌کنند.

	P1	P2	P3	P4
1	0	0	1	0
2	1	0	0	0
3	0	0	0	1
4	0	1	0	0

شکل(۱): یک نمونه ماتریس موقعیت

۳-۲ سرعت ذره‌ها

سرعت هر ذره نیز به صورت یک ماتریس $m * m$ در نظر گرفته شده است که عناصر آن در بازه $[-V_{max}, V_{max}]$ قرار دارند. همچنین $gbest$ و $pbest$ نیز چون از جنس مکان و موقعیت ذره‌ها هستند، به عنوان ماتریس‌های $m * m$ در نظر گرفته شده است که هر عنصر آن یا صفر است یا یک. در الگوریتم PSO پیشنهادی، از هم‌بندی ستاره‌ای برای شبیه‌سازی رابطه همسایگی استفاده شده است. در هر مرحله زمانی از اجرای الگوریتم PSO پیشنهادی، $gbest$ و $pbest$ با توجه به مقدار برازندگی ذره‌های نسل فعلی بروز می‌شوند، مقدار برازندگی هر ذره (مثلاً x_i) تخمین زده می‌شود و اگر مقدار آن بزرگتر باشد از مقدار برازندگی بهترین موقعیت آن ذره ($pbest_i$)، آن‌گاه $pbest_i$ جایگزین x_i می‌شود(با انجام دادن این عملیات سعی داریم که ذره‌های هر نسل بهتر از نسل قبل باشند و از بدتر شدن ذره‌ها در نسل‌های بعدی جلوگیری می‌کنیم).

این نکته نیز قابل توجه است که $gbest$ بهترین $pbest$ است، به عبارت دیگر $gbest$ یکی از $pbest$ ها است که کمترین مقدار برازندگی (min T) را دارد و در نهایت نیز الگوریتم PSO پیشنهادی، $gbest$ را به عنوان پاسخ نهایی و بهینه برمی‌گرداند.

۳-۳ بروز نمودن ذره‌ها

در الگوریتم PSO پیشنهادی معادله شماره (۱) برای بروز کردن ماتریس سرعت ذره‌ها و معادله شماره (۲) برای بروز نمودن ماتریس موقعیت ذره‌ها استفاده می‌شود[۴].

شده، در اکثر موارد جواب‌های بهتر و نزدیک‌تری به جواب بهینه تولید می‌کند. این نتیجه‌گیری حتی برای مقادیر بزرگ m و مقادیر نزدیک به یک و بزرگتر از یک برای پارامتر دلتا نیز صادق است.

مراجع

[] [منصفی، رضا، حمیدزاده، جواد، " زمان‌بندی بار محاسباتی تقسیم‌پذیر با در نظر گرفتن زمان بازگشت نتایج در سیستم‌های ناهمگن با استفاده از الگوریتم ژنتیک"، چهاردهمین کنفرانس انجمن کامپیوتر، دانشگاه صنعتی امیرکبیر، اسفند ماه ۱۳۸۷، تهران- ایران.

[2] Bharadwaj, V., Ghose, D., Mani, V., Robertazzi, T. G., *Scheduling Divisible Loads in parallel and Distributed Systems*, IEEE CS Press, 1996.

[3] Vanderbei, R. J., *Linear Programming: Foundations and Extensions*, 2nd Ed., International Series in Operations Research & Management, vol. 37, Kluwer Academic Publishers, 2001.

[4] Hesam Izakian, Behrouz Tork Ladani, Kamran Zamanifar, Ajith Abraham, "A Novel Particle Swarm Optimization Approach for Grid Job Scheduling".

[5] Ajith Abraham, Hongobo Liu, Weishi Zhang, Tae-Gyu Chang, "Scheduling Jobs On Computational Grids Using Fuzzy Particle Swarm Algorithm", 2006.

[6] Haupt, R. L., Haupt, S. E., *Practical Genetic Algorithms*, 2nd Ed., John Wiley & Sons, 2004.

[7] Bharadwaj, V., Ghose, D., Robertazzi, T. G., "Divisible Load Theory: A New Paradigm for Load Scheduling in Distributed Systems", *Cluster Computing*, vol.6, no.1, pp.7-17, jan. 2003.

[8] Robertazzi, T. G., "Ten Reasons to Use Divisible Load Theory", *Computer*, pp. 63-68, May 2003.

[9] Jingxi, J., Bharadwaj, V., Ghose, D., "Adaptive Load Distribution Strategies for Divisible Load Processing on Resource Unaware Multilevel Tree Networks", *IEEE Transactions on Computers*, vol. 65, no. 7, pp. 999-1005, 2007.

[10] Ghatpande, A., Nakazato, H., Beaumont, O., Watanabe, H., "SPORT: An Algorithm for Divisible Load Scheduling With Result Collection on Heterogeneous Systems", *IEICE Transactions on Communications*, vol.E91-B, no.8 August 2008.

[11] Ghatpande, A., Nakazato, H., Beaumont, O., Watanabe, H., "Analysis of Divisible Load Scheduling with Result Collection on Heterogeneous Systems", *IEICE Transactions on Communications*, vol. E91-B, no. 7, July 2008.

[12] S. N. Omkar, Manoj Kumar. M, Department of Aerospace Engineering, Indian Institute of Science, Bangalore, 560012, KA, India, " Optimal Load Scheduling in a Single Level Three Network with Communication Start-Up Delays Using Particle Swarm Optimization".

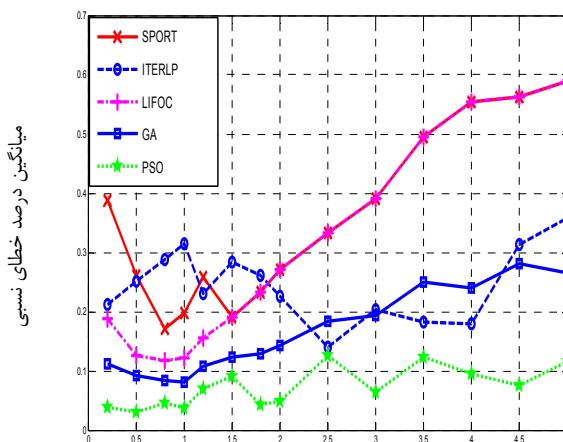
[13] Cheng, Y. C., Robertazzi, T. G., "Distributed Computation with Communication Delays", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 24, no. 6, pp. 700-712, Nov. 1988.

[14] Rosenberg, A. L., "Sharing Partitionable Workloads in Heterogeneous NOWs: Greedier Is not Better", *IEEE International Conf. on Cluster Computing*, pp. 124-131, Newport Beach, CA, Oct. 2001.

[15] Ghatpande, A., Nakazato, H., Watanabe, H., Beaumont, O., "Divisible Load Scheduling with Result Collection on Heterogeneous Systems", *Proc. Heterogeneous Computing Workshop(HCP 2008)*, April 2008.

[16] Beaumont, O., Marchal, L., Rehn, V., Robert Y., "FIFO Scheduling of Divisible Loads with Return Messages Under the One Port Model", *Proc. Heterogeneous Computing Workshop HCW'06*, April 2006.

در نمودار شکل (۲)، میانگین درصد خطای نسبی برای پنج الگوریتم PSO، ژنتیک، LIFO، ITERLP و SPORT و برای پنج کامپیوتر کارگر ترسیم شده است. همان طور که در نمودار شکل (۲) مشاهده می‌شود برای تمام مقادیر دلتا، میانگین درصد خطای نسبی الگوریتم PSO در مقایسه با سایر الگوریتم‌ها کمتر است.



شکل(۲): نمودار میانگین درصد خطای نسبی برای $m=5$

الگوریتم PSO نسبت به الگوریتم ژنتیک پاسخ‌های بهتری را تولید نموده و زمان پاسخ بهتری نیز دارد بنابراین کارایی الگوریتم PSO بهتر از کارایی الگوریتم ژنتیک است. در جدول شماره (۲)، الگوریتم‌ها از نظر میانگین مدت زمان اجرا و میانگین درصد خطای نسبی با یکدیگر مقایسه شده‌اند.

جدول(۲): مقایسه زمان اجرا و میانگین کل درصد خطای

نسبی الگوریتم‌ها برای $m=5, \delta=0.5$

نام الگوریتم	میانگین زمان اجرا (ثانیه)	میانگین درصد خطای نسبی
الگوریتم بهینه	440.657	0
الگوریتم PSO	32.7028	0.0301187
الگوریتم ژنتیک	35.7704	0.0927642
الگوریتم ITERLP	1.5581	0.252107
الگوریتم LIFO	0.029952	0.126484
الگوریتم SPORT	0.001872	0.260995

- نتیجه‌گیری و کارهای آینده

مسئله زمان‌بندی بار محاسباتی تقسیم‌پذیر با در نظر گرفتن زمان بازگشت نتیجه در یک سیستم ناهمگن، جزو مسائل ترکیباتی با درجه پیچیدگی بالا و از مرتبه زمانی $O(m!^2)$ است که هنوز الگوریتمی بهینه از مرتبه چند جمله‌ای برای حل آن ارائه نشده است. روش‌های موجود نیز روش‌هایی ابتکاری هستند که نمی‌توانند در تمامی حالات، جواب قابل قبول تولید کنند. در این میان، الگوریتم PSO پیشنهاد

