

ترکیب آتاماتای مهاجرت اشیا و الگوریتم ژنتیک برای زمانبندی گراف وظایف در معماری چند پردازنده ای

سعید پارسا، حبیب ایزدخواه و امیر حسین زاده

آدرس محل کار: دانشگاه علم و صنعت ایران

E-mail: pارسا@iust.ac.ir, habib_eizadkhah@yahoo.com, amirhosseinzade@yahoo.com

چکیده - امروزه سیستمهای چندپردازنده ای کاربرد وسیعی در محاسبات موازی دارند. در این سیستمها زمانبندی مؤثر برای اجرای یک برنامه موازی جهت نائل شدن به کارآیی بالا امری حیاتی است. این زمانبندی باید به گونه ای انجام گیرد که بتواند زمان اجرای کل برنامه را با توجه به زمان وظایف و ارتباط بین پردازنده ها، کمینه نماید. با توجه به NP-Hard بودن مسئله زمانبندی گراف وظایف، رویکرد های مبتنی بر روشهای قطعی در این زمینه کارا نخواهند بود؛ بنابراین استفاده از پردازش تکاملی و به طور عمده الگوریتمهای ژنتیک و الگوریتم های ترکیبی برای حل این مسئله مؤثر می باشد. با ترکیب الگوریتم ژنتیکی و آتاماتای یادگیر و تلفیق مفاهیم ژن، کروموزوم، اقدام و عمق، می توان به یک روش جستجوی کارا برای حل مساله گراف وظایف دست یافت، بطوریکه با استفاده هم زمان از آتاماتای یادگیر و الگوریتم ژنتیک در فرآیند جستجو، سرعت رسیدن به جواب، افزایش چشم گیری پیدا می کند و از بدام افتادن الگوریتم در حداقل های محلی جلوگیری می شود. الگوریتم پیشنهادی در این مقاله کوششی است در جهت خودترمیمی، تولید مثل، جرمه و پاداش (هدایت) که از ویژگی های مهم الگوریتم ترکیبی است.

رویکرد جدید در این الگوریتم علاوه بر ترکیبی بودن الگوریتم، بر پایه کوتاهتر کردن طول مسیر بحرانی و کاهش هزینه ارتباطات بین پردازنده ای است. در نهایت نتایج عملی حاصل از پیاده سازی روش ارائه شده نشان می دهد که می توان یک زمانبندی مناسب در زمان بسیار کمتری نسبت به الگوریتمهای مشابه پیدا کرد.

کلید واژه - زمانبندی چند پردازنده ای، گراف وظایف و الگوریتمهای ژنتیک.

۱- مقدمه

در رویکرد های قبلی برای حل مسئله زمانبندی گراف وظایف روی معماری چندپردازنده ای چند الگوریتم غیرژنتیک پیشنهاد شده است. در این زمینه شش الگوریتم غیرژنتیک مهم [۴] وجود دارد که عبارتند از: ETF [۵]، DLS [۶]، LAST [۷]، HLFET [۸]، ISH [۹] و MCP [۱۰]. از بین این الگوریتمها، MCP بهترین الگوریتم غیرژنتیک برای حل این مسئله می باشد و به همین علت برای ارزیابی و مقایسه الگوریتمهای دیگر استفاده می گردد [۴]. الگوریتمهای قدیمی سعی در ساده تر کردن فرضیات خود درباره گراف وظایف داشتند. به طور مثال برخی از الگوریتمها فرض می کنند هزینه های محاسباتی تمام وظایف یا کارها یکسان است و بعضی دیگر این هزینه ها را اختیاری در نظر می گیرند. تعدادی از الگوریتمها، وزن یاها، یعنی هزینه ارتباطات بین پردازنده ها را برابر با صفر فرض می کنند [۴]. تعداد پردازنده ها نیز در برخی الگوریتمها محدود و در برخی

امروزه سیستمهای چندپردازنده ای کاربرد وسیعی در محاسبات موازی دارند. در این سیستمها زمانبندی مؤثر برای اجرای یک برنامه موازی جهت نائل شدن به کارآیی بالا امری حیاتی است. این زمانبندی باید به گونه ای انجام گیرد که بتواند زمان اجرای کل برنامه را با توجه به زمان وظایف و ارتباط بین پردازنده ها، کمینه نماید. مسئله زمانبندی گراف وظایف یک مسئله NP-Hard [۳] است. بنابراین رویکرد های مبتنی بر روشهای قطعی در این زمینه کارایی چندانی نخواهند داشت. استفاده از الگوریتمهای پردازش تکاملی و به طور عمده الگوریتمهای ژنتیک و الگوریتمهای پردازش تکاملی ترکیبی با توجه به ماهیت غیر قطعی که دارند؛ برای حل این مسئله مؤثر می باشد.