



اجباری - مهلت تحویل ۱۳ بهمن

هدف از این پروژه، آشنایی عملی با روال هم طراحی سخت افزار و نرم افزار در قالب پیاده سازی یک نمونه عملیاتی در سطح سیستم می باشد. بدین منظور، طراحی توأم سخت افزاری و نرم افزاری الگوریتم بهینه سازی استراتژی تکاملی (Evolutionary Strategy) به عنوان یکی از الگوریتم های بهینه سازی فرامکاشفه ای پر کاربرد در نظر گرفته شده است. هدف از این طراحی، بکارگیری موثر این الگوریتم در بستر سیستم های نهفته و افزایش کارایی این روش بهینه سازی در کنار در نظر داشتن ملاحظات هزینه و فضای پیاده سازی می باشد. بدین منظور اجزای اصلی این الگوریتم در قالب بلوک های عملیاتی در نظر گرفته شده و پیاده سازی آن ها در بستر سخت افزار یا نرم افزار مورد بحث قرار می گیرد.

الگوریتم استراتژی های تکاملی، روش بهینه سازی و جستجوی مبتنی بر جمعیت و خود تطبیق پذیر است که با الهام از پدیده انتخاب طبیعی (natural selection) ارائه شده است. این الگوریتم از لحاظ پیاده سازی و ایده اولیه ساده می باشد و در زمان مشخص جواب نهایی مناسبی فراهم می آورد و در نتیجه انتخاب خوبی برای بسیاری از کاربردهای مهندسی از جمله سیستم های نهفته می باشد. با وجود ساده تر بودن این الگوریتم نسبت به سایر روش های بهینه سازی تکاملی، همچنان بکارگیری آن در بستر سیستم های نهفته به دلیل محدودیت های ذاتی آن ها در توان پردازشی، حجم حافظه و ... چالش برانگیز می باشد. در نتیجه با طراحی موثر سعی می شود تا این الگوریتم با کارایی مناسب پیاده سازی شود و زمان پردازشی آن متناسب باشد.

طراحی ماژولار این الگوریتم و بخش بندی و توزیع آن بین اجزای پردازشی سخت افزاری و نرم افزاری، یکی از موثرترین روش ها در بهبود کارایی و بکارگیری آن در سیستم های نهفته می باشد. بدین منظور تصمیم درباره تخصیص هر ماژول به سخت افزار یا نرم افزار بر حسب الزامات مسئله و سیستم متفاوت است. در این پروژه ابتدا یکی از تقسیم بندی های ماژولار مناسب انتخاب شده و هدف آن است که براساس آن، روال هم طراحی این الگوریتم بر بستر سخت افزار و نرم افزار انجام گیرد. بدین ترتیب در نهایت سیستمی خواهیم داشت که قادر است الگوریتم ژنتیک را با کارایی و هزینه مناسب پیاده سازی کند و در ساختار سیستم های نهفته بکار گرفته شود.

کلیات این الگوریتم و روال کار آن در ادامه توضیح داده می شود و توصیه می گردد به منظور آشنایی بیشتر با روال کار این الگوریتم مطالعه بیشتر در این حیطه انجام گیرد. در ساختار پایه این الگوریتم، کار با جمعیتی از جواب های

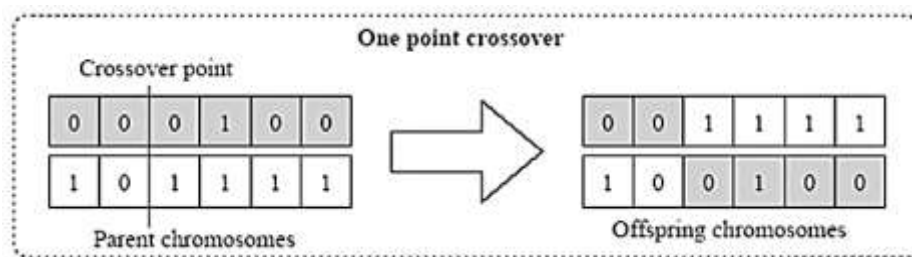


اجباری - مهلت تحویل ۱۳ بهمن

کандید مسئله که به صورت تصادفی تولید شده اند آغاز می شود (جمعیت اولیه). هریک از جواب های تصادفی یک کروموزوم نامیده می شوند و در ادامه با اعمال تغییر بر روی آن ها جمعیت را به سوی جواب بهینه سوق می دهیم. علمگرهای این الگوریتم در حلقه بکار گرفته می شوند و هر تکرار حلقه یک نسل (generation) را شکل می دهد. اجرای حلقه ها و تولید نسل ها تا رسیدن به شرط خاتمه الگوریتم ادامه می یابد. شکل زیر ساختار جمعیت مدنظر در این الگوریتم را نشان می دهد.



هدف از تغییرات تدریجی در جمعیت، حذف موارد نامناسب و تکثیر موارد نزدیک به بهینه است. پس از تولید جمعیت اولیه، نوبت به تولید نسل های بعدی در حین تکرارها می رسد. بدین منظور دو عضو بصورت تصادفی و متناسب با شایستگی به عنوان والدین انتخاب می شوند. پس از اعمال recombination بین والدین و تولید λ فرزند، آن ها به نسل قبل اضافه شده و بهترین آن ها برای نسل بعد انتخاب می شوند. بدین منظور یک نقطه از هر والد به طور تصادفی انتخاب می شود و دو عضو شایسته انتخاب شده از این نقطه شکسته شده و با باز ترکیبی آن ها دو فرزند ایجاد می شوند. نمایی از این عملیات در شکل زیر نشان داده شده است. این عملیات با احتمال مشخصی (P_{co}) انجام می شود.



همچنین به منظور ایجاد تنوع مناسب در جمعیت با هدف جستجوی موثرتر فضای طراحی، در این الگوریتم عمگر جهش (mutation) نیز در نظر گرفته شده است که رسیدن به نقطه جدید را تسهیل می کند. عملیات جهش با افزودن یک نویز گاوسی با میانگین صفر (نویز سفید) به جمعیت مورد نظر انجام می شود. به این منظور برای هر



اجباری - مهلت تحویل ۱۳ بهمن

بعد مسئله (متغیر تصمیم)، یک مقدار انحراف از معیار در نظر گرفته می شود (σ_j برای بعد z_j). متغیرهای هر نمونه تحت جهش به شکل زیر تغییر می کنند:

$$x_j \rightarrow x_j + \sigma_j \cdot N(0,1)$$

که در این رابطه، $N(0,1)$ یک متغیر تصادفی از توزیع گاوسی با میانگین صفر و واریانس ۱ است. پس از اعمال رابطه فوق بر روی همه ابعاد همه نمونه ها (فرض احتمال جهش برابر ۱) انحراف از معیار با بکارگیری رابطه زیر اصلاح می شود:

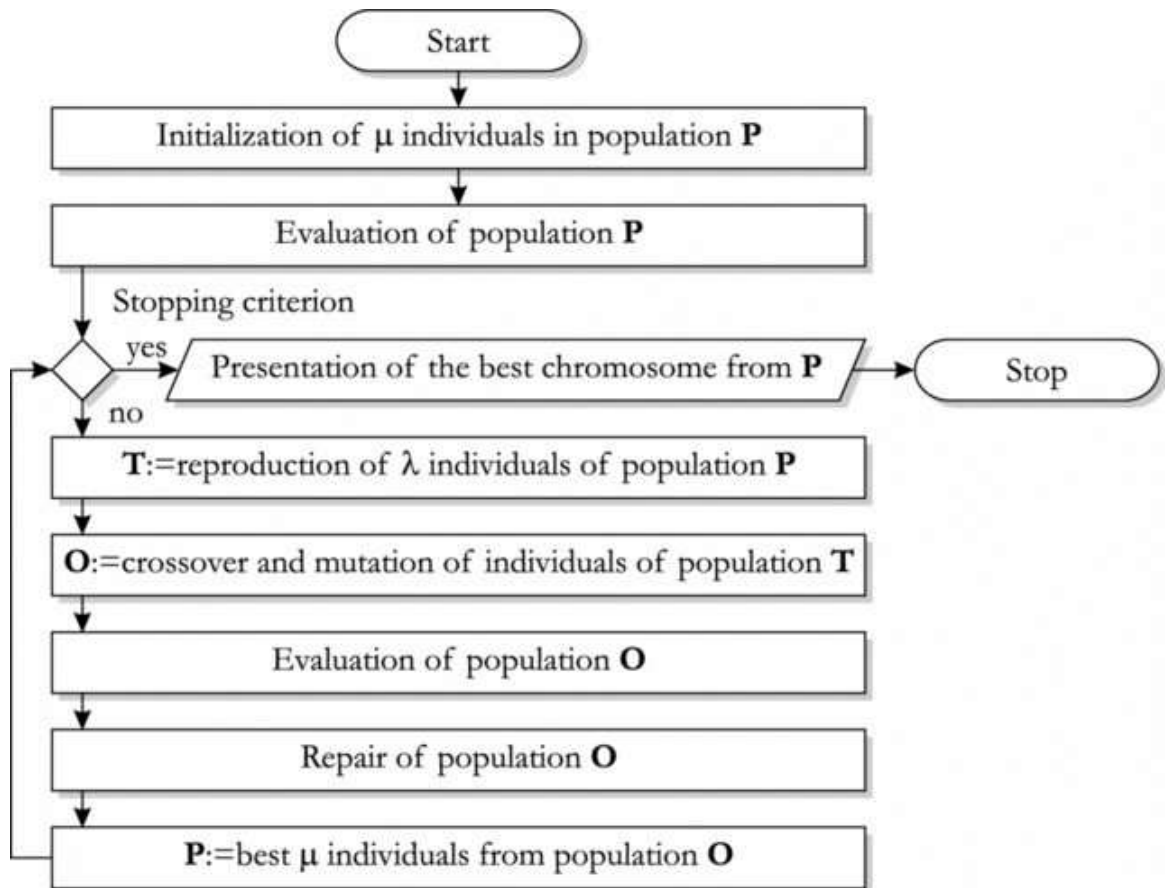
$$\sigma_j \rightarrow \sigma_j \cdot e^{(N(0,1) - N_j(0,1))}$$

در این رابطه، $N(0,1)$ یک متغیر تصادفی با میانگین صفر و انحراف از معیار یک می باشد که برای همه ابعاد مشترک است و N_j به ازای هر بعد تولید می شود.

پس از این مرحله، بهترین والدین و فرزندان به نسل بعدی انتقال داده می شوند و ارزیابی روی آن ها انجام می گیرد و اگر شرط پایان الگوریتم برقرار نشود، روال شرح داده شده ادامه می یابد. شرط پایان این الگوریتم برحسب معیارهای مختلف مانند تعداد تکرار معین، رسیدن به جواب به اندازه کافی خوب که توسط کاربر مدنظر بوده است یا همگرایی شایستگی تنظیم می گردد. فلوچارت مراحل الگوریتم در شکل زیر نشان داده شده است.



اجباری - مهلت تحویل ۱۳ بهمن



پروژه پیش رو با هدف پیاده سازی الگوریتم استراتژی تکاملی در فرایند طراحی توأم سخت افزار و نرم افزار تعریف شده است. به منظور انجام این پروژه لازم است ابتدا براساس توضیحات ارائه شده و مطالعات اضافه، تسلط کافی بر این الگوریتم جستجوی فضا داشته باشید و سپس اجزای مازول های اصلی و پروسه های سیستمی که قصد پیاده سازی آن را در طی این پروژه دارید را مشخص نمایید. برای استفاده از حداکثر موازی سازی قصد داریم در طی پیاده سازی، عملیات را بین مازول های سخت افزاری و نرم افزاری افراز نمائیم. بدین منظور لازم است بخش سخت افزاری و بخش نرم افزاری را به صورت مناسب پیاده سازی کرده و ارتباط مناسبی بین این دو بخش برقرار نمایید. به منظور پیاده سازی ساختار سیستم از زبان توصیف SystemC استفاده کنید و با برقراری اتصال صحیح بین مازول های سخت افزاری و نرم افزاری، طراحی خود را تکمیل نمایید. ساختار شرح داده شده با هدف حل مسائل



اجباری - مهلت تحویل ۱۳ بهمن

بهینه سازی تک هدفی مختلف مورد استفاده قرار می گیرد و ورودی آن ساختار کروموزوم های جمعیت براساس تعریف و نوع مسئله می باشد.

در نهایت به منظور ارزیابی و بررسی درستی ساختار پیاده سازی شده، مسئله محک زیر را توسط ساختار پیاده سازی شده حل کنید و نتیجه نهایی را ارائه نمایید.

«فرض کنید قصد داریم مسئله کوله پشتی را توسط سیستم توسعه داده شده بر مبنای الگوریتم استراتژی تکاملی حل کنیم. اگر بدانیم اشیای کاندید شده مشخصاتی مطابق جدول زیر دارند و سود حاصل از هر وسیله را نیز توسط تقسیم ارزش به وزن آن به دست آوریم و امکان برداشتن درصدی از هر شی نیز وجود دارد، بهترین سود حاصل از برداشتن این اشیاء را توسط سیستمی که پیاده سازی کرده اید بیابید»

Item	1	2	3	4	5	6	7	8	9
Price	6	5	8	9	6	7	3	6	8
Weight	2	3	6	7	5	9	3	4	5

خروجی می بایست شامل گزارشی از روال انجام کار و نتایج به دست آمده از ارزیابی طراحی انجام شده به همراه کدهای پیاده سازی شده باشد. اندازه جمعیت، احتمال باز ترکیبی و جهش را به صورت دلخواه در نظر بگیرید و شرط خاتمه را به صورت تغییر نکردن جواب بهینه در ۵ تکرار متوالی تنظیم کنید. (توجه داشته باشید که ۵ تکرار اولیه را برای شرط خاتمه معیار قرار ندهید). همچنین جمعیت اولیه را به صورت تصادفی تولید کنید.

بخش امتیازی:

پیاده سازی بخش نرم افزاری سیستم را در محیط Matlab انجام دهید و با برقراری اتصال صحیح بین بخش های سخت افزاری (شبیه سازی شده در محیط SystemC) و نرم افزاری (شبیه سازی شده در محیط Matlab)، طراحی خود را تکمیل نمایید و توسط مسئله محک داده شده درستی عملکرد سیستم را ارزیابی و نسبت به حالت قبل مقایسه نمایید.

"موفق باشید"