

گزارش بخش ژنتیک

حسام اسداله زاده – 810198346

تعریف ژن و کروموزوم: هر ژن یک آرایه 9 تایی از اعداد 1 تا 9 بوده و هر کروموزوم شامل 9 ژن می باشد. هنگام خواندن جدول سودوکو از فایل برنامه طوری طراحی شده که اعداد هر Subgrid را در یک ژن ذخیره کند. یعنی هر ژن شامل یک سطر یا ستون از فایل تست نبوده و در واقع شامل یک زیرجدول از جدول اصلی است. دلیل این انتخاب این است که در هنگام تولید جمعیت اولیه اطمینان داریم که برنامه حداقل دارای حل درست از جهت زیرجدول ها می باشد.

جمعیت اولیه: کروموزوم هر عضو جمعیت اولیه شامل 9 ژن است که هر کدام از این ژن ها یک جایگشت تصادفی از اعداد 1 تا 9 هستند. ولی بعد از اعداد ثابت جدول که در ورودی به ما داده شده اند از آن ژن حذف شده و در مکان ثابت خود قرار می گیرند. بدین ترتیب به تعداد `population_size` کاندید اولیه به عنوان جمعیت اولیه تولید می شود.

تابع معیار سازگاری: حاصل جمع تعداد اعداد تکراری در هر سطر و ستون می باشد. بدیهی است که با تعریف ما از ژن، عدد تکراری در زیرجدول وجود نخواهد داشت و هرگاه تعداد اعداد تکراری در هر سطر و ستون برابر 0 شد یعنی به جواب صحیح رسیدیم.¹

عملیات crossover: یک عدد تصادفی از 0 تا 7 انتخاب شده و جای اعداد مکان تصادفی و مکان بعدی آن در دو کاندید با هم عوض می شود. دقت کنیم که هر دو عدد انتخاب شده در زیرجدول یکسانی قرار دارند.

عملیات mutation: با احتمال `mutationrate` جای دو عدد در یک زیرجدول یک کاندید با هم عوض می شود. توجه کنید این عمل در صورتی انجام می شود که جای عدد در جدول اولیه ثابت نبوده و جایگاه قابل تغییر باشد.

عملیات selection: همه ی اعضای جمعیت با تابع معیار سازگاری رتبه بندی شده و مرتب می شوند و درصدی از آنها به نسل بعد منتقل می شوند (با نرخ selectionrate)²

پارامتر دیگری به نام maxgencount داریم که حداکثر نسل های تولید شده را مشخص می کند. اگر مسئله به تعداد ماکسیمم نسل رسید و هیچ جواب با فیتنس 0 پیدا نکرد به پایان رسیده و مسئله بی پاسخ باقی می ماند.

پاسخ سوال 3: احتمال mutation را می توانیم برابر 0.08 تا 0.1 انتخاب کنیم که بنابر تجربه نتایج خوبی می دهد. اگر مقدار بیشتر از این باشد ممکن است همواره پاسخ نزدیک به پاسخ صحیح جهش یابد و از حالت مطلوب دورتر شود.

پاسخ سوال 4: برای جلوگیری از این اتفاق می توان اندازه جمعیت اولیه را بزرگتر در نظر گرفت و یا عملیات mutation و crossover را هوشمندانه تر انجام داد به طوری که همواره فیتنس کروموزوم نهایی بیشتر شود. و یا می توان مقادیر پارامترهای مسئله را متغیر در نظر گرفت و با توجه به شرایط آنها را در حین برنامه تغییر داد.