رمزنگاری متن مبتنی بر گلایدرها در تئوری بازی زندگی

مجید وفایی جهان، فائزه خسروجردی
گروه کامپیوتر نرم افزار، دانشگاه آزاد اسلامی مشهد، VafaeiJahan@mshdiau.ac.ir
گروه کامپیوتر نرم افزار، دانشگاه آزاد اسلامی مشهد، Faezeh.khosrojerdi@gmail.com

چکیده - یکی از معروف‌ترین قوانین در اتوماتای سلولی بازی زندگی است که قادر به تولید قالب‌هایی متفاوت و با ویژگی‌هایی خاص است. در این میان گلایدرها قالب‌هایی متحرک و متناوب بوده که می‌توانند توسط قالب متناوب تفنگ گلایدر در شبکه سلولی تولید شده و توسط قالب ثابت خورنده از بین بروند. همچنین برخورد بین دو رشته گلایدر می‌تواند باعث نابودی آن‌ها شود. این مجموعه ویژگی‌های گلایدرها در شبیه‌سازی رفتار گیت‌ها و توابع منطقی استفاده می‌شود. به‌این‌ترتیب امکان پیاده‌سازی تمامی مسائل دودویی در فضای اتوماتای سلولی، به‌کمک گلایدرهای بازی زندگی فراهم می‌شود. در این مقاله با نگاشت متن به فضای دودویی و استفاده از توابع منطقی پیاده‌سازی شده به‌کمک گلایدرها، به‌عنوان یک تابع رمزنگاری، به معرفی و تشریح یک روش رمزنگاری متن مبتنی بر گلایدرهای بازی زندگی پرداخته شده است.

کلید واژه- اتوماتای سلولی، بازی زندگی، گلایدر، تفنگ گلایدر، خورنده.

۱- مقدمه

اتوماتای سلولی یک مدل ریاضی است، که می‌تواند برای محاسبات و شبیه‌سازی سیستم‌ها بکار رود. این سیستم‌ها با قوانین ساده و محلی محاسبات و رفتار پیچیده‌ای از خود بروز می‌دهند. هر سلول مجموعه‌ای متناهی از وضعیت‌ها دارد، که در هر لحظه باتوجه به حالت خود و همسایه‌هایش تغییر وضعیت می‌دهد [۱۰]. مجموعه اتوماتای سلولی را می‌توان براساس رفتار آن‌ها به چهار کلاس تقسیم‌بندی کرد. در کلاس ۱ و با هر نوع پیکربندی اولیه، اتوماتا بعد از چند مرحله تکامل به یک وضعیت همگن می‌رسد، که در آن تمامی سلول‌ها دارای وضعیت یکسانی هستند. در کلاس ۲، اتوماتا بعد از چند مرحله تکامل ساختارهای ساده‌ای را تولید می‌کند، که برخی از آن‌ها ثابت و برخی دارای رفتار متناوبی هستند. در کلاس ۳، تکامل اتوماتا روند مشخصی را دنبال نمی‌کند و در شبکه سلولی بی‌نظمی به‌وجود می‌آید. سایر اتوماتاها که متعلق به این سه کلاس نباشند، در کلاس ۴ جای می‌گیرند [۶،۵]. در میان اعضای دو بعدی و دو وضعیتی از کلاس ۴، بازی زندگی به‌عنوان اتوماتایی شناخته می‌شود که کاندید مناسبی برای شبیه‌سازی رفتار بسیاری از سیستم‌های پیچیده می‌باشد؛ به شکلی که کانوی توانایی آن را در شبیه‌سازی ماشین تورینگ با استفاده از گلایدر و تفنگ گلایدر ثابت کرد [۱۰،۶]. گلایدرها قالب‌های متناوبی هستند، که در طول دوره تناوب خود در فضای سلولی حرکت می‌کنند و سپس به وضعیت اولیه خود بازمی‌گردنند. تفنگ گلایدر رشته‌ای از گلایدرها را در شبکه سلولی ساطع می‌کند، که این رشته می‌تواند حاوی اطلاعات باشد و آن‌ها را جابجا کند. برخورد بین گلایدرها در فضای سلولی می‌تواند باعث نابودی آن‌ها شود و از نتایج حاصل از آن می‌توان در شبیه‌سازی رفتار گیت‌ها و توابع منطقی بهره گرفت [۶،۵].

در این مقاله ابتدا با معرفی مختصری از اتوماتای سلولی و بازی زندگی به تشریح گلایدر و ویژگی‌های آن به‌عنوان یکی از قالب‌های بازی زندگی پرداخته شده است. در ادامه پیاده‌سازی گیت‌ها و توابع منطقی با استفاده از این قالب بررسی شده است. در انتها یک روش رمزنگاری متن برپایه ویژگی‌های گلایدرها و توابع منطقی پیاده‌سازی شده به‌وسیله آن‌ها، معرفی می‌شود.

۲- اتوماتای سلولی

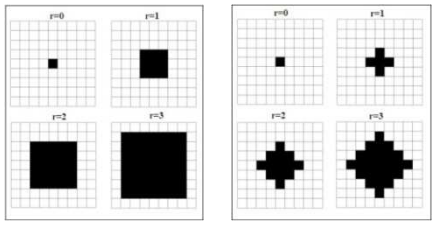
اتوماتای سلولی سیستم‌های دینامیکی گسسته‌ای هستند، که رفتارشان کاملا براساس ارتباط محلی استوار است. این سیستم‌ها متشکل از سلول‌هایی هستند، که هر یک از آن‌ها در هر مرحله، تنها در یک حالت از مجموعه‌ای از حالت‌های متناهی قرار دارند. زمان در اتوماتای سلولی به‌صورت گسسته

پیش می‌رود و از طریق اعمال قوانین سراسری در هر مرحله، سلول‌ها وضعیت جدید خود را به‌دست می‌آورند [۱۰،۱]. با توجه به ابعاد شبکه سلولی، اتوماتا می‌تواند در ابعاد متفاوتی تعریف شود؛ با این‌وجود انواع یک بعدی، دو بعدی و سه بعدی مرسوم‌تر هستند. در این مقاله نیز از اتوماتای سلولی دو بعدی و دو وضعیتی استفاده شده است.

یک اتوماتای سلولی دو بعدی به صورت2D-CA نشان داده می‌شود؛ و می‌توان آن را با چندتایی CA=(I,N,V,F) تعریف کرد. در این چندتایی، I فضای سلولی است که به‌صورت آرایه‌ای دو بعدی در ابعاد r×c پیکربندی می‌شود؛ بنابراین می‌توان آرایه I را با استفاده از رابطه شماره (۱) تعریف کرد [۱۱]. همچنین در چندتایی مذکور، N نوع همسایگی را تعیین می‌کند و V یک مجموعه متناهی از وضعیت‌های ممکن سلولی است. همچنین F، قانون انتقال محلی را مشخص می‌کند.

$$I = \{(a,b) : 1 \le a \le r, 1 \le b \le c\} \quad (۱)$$

قوانین اتوماتای سلولی نحوه تاثیر پذیرفتن سلول از سلول‌های همسایه را مشخص می‌کنند. یک سلول، همسایه سلول دیگر نامیده می‌شود اگر بتواند آن سلول را در یک مرحله و براساس قانون حاکم، تحت تاثیر قرار دهد. ویژگی‌های اساسی اتوماتای سلولی، فضای گسسته، زمان گسسته، محدودیت تعداد وضعیت‌های ممکن، یکسان بودن تمام سلول‌ها، قطعی بودن قوانین و وابستگی قانون در هر سلول به مقادیر سلول‌های همسایه آن می‌باشد [۱۱]. از میان انواع همسایگی‌های موجود، دو حالت متداول با شعاع‌های متفاوت برای اتوماتای سلولی دو بعدی در شکل ۱ آورده شده است.



(ب) (الف)

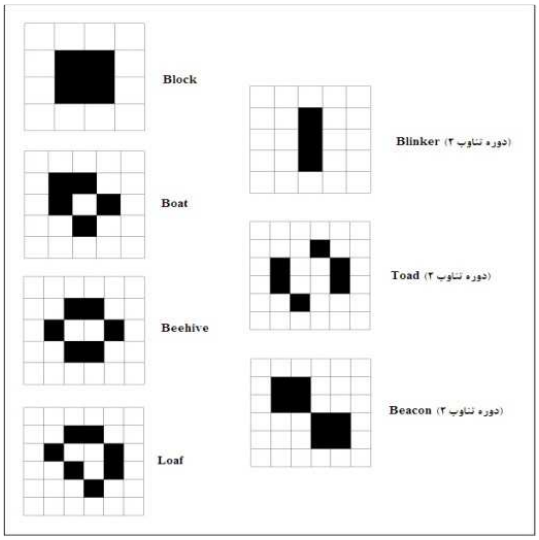
شکل ۱: (الف) همسایگی ون نیومن (ب) همسایگی مور

۳- بازی زندگی

بازی زندگی اولین بار در اکتبر ۱۹۷۰ در ستون بازی و ریاضی مجله علمی Scientific American، توسط یک ریاضی‌دان انگلیسی به نام جان هورتون کانوی مطرح شد [۲]. این زندگی، یک اتوماتای سلولی دو بعدی است، که از همسایگی مور استفاده می‌کند. هر سلول می‌تواند یکی از دو حالت (زنده=۱) و یا (مرده=۰) را داشته باشد. در هر مرحله از بازی، سلول‌های شبکه برمبنای این قوانین انتقال به‌روزرسانی می‌شوند: **(۱)**: هر سلول زنده با کم‌تر از ۲ همسایه زنده، می‌میرد. (۲): هر سلول زنده با بیش از ۳ همسایه زنده، می‌میرد. (۳): هر سلول زنده با ۲ و یا ۳ همسایه زنده، زنده باقی می‌ماند و به نسل بعد می‌رود. (۴): هر سلول مرده با دقیقا ۳ همسایه زنده، زنده می‌شود [۱۲،۶].

۳-۱- قالب‌های بازی زندگی

اجرای بازی زندگی با یک پیکربندی اولیه و تصادفی می‌تواند قالب‌های متفاوتی را به‌وجود آورد. بعضی از این پیکربندی‌ها، در ادامه بازی بدون تغییر باقی می‌مانند که قالب‌های ثابت نامیده می‌شوند. دسته دوم نوسان‌گرها هستند که رفتار متناوبی از خود نشان می‌دهند و بین یک تعداد متناهی از پیکربندی‌ها نوسان می‌کنند که تعداد این پیکربندی‌ها دوره تناوب آن را مشخص می‌کند [۱۲]. تعدادی از این پیکربندی‌های ثابت و نوسان‌گر در شکل ۲ دیده می‌شود [۱۸،۲].

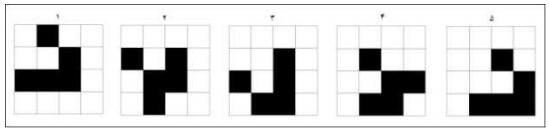


شکل ۲: چند نمونه از قالب‌های نوسان‌گر و ثابت در بازی زندگی

۴- گلایدر

یکی از پیکربندی‌های نوسان‌گر در بازی زندگی گلایدرها می‌باشند. گلایدرها قالب‌هایی هستند که در طول یک دوره تناوب به‌طور قطری در شبکه سلولی حرکت می‌کنند [۱۴] و سپس به جهت اولیه خود بازمی‌گردند و با شروع دوره تناوب بعد، این روند تکرار می‌شود [۱۵،۱۲]. بنابراین ویژگی مهم گلایدرها متحرک بودن آن‌ها در شبکه سلولی است. در شکل ۳ یک گلایدر با دوره تناوب ۴ نشان داده شده است [۹]، که در

ادامه مقاله نیز برای پیاده‌سازی از همین نوع گلایدر استفاده می‌شود.



شکل ۳: حرکت گلایدر در یک دوره تناوب و بازگشت به جهت اصلی

۴-۱- تفنگ گلایدر

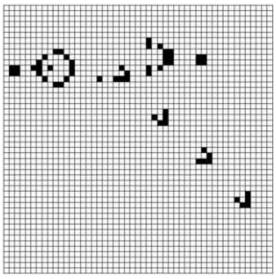
یکی دیگر از قالب‌های بازی زندگی تفنگ گلایدر است، که به‌عنوان مولد گلایدر در بازی زندگی عمل می‌کند. تفنگ گلایدر یک قالب نوسان‌گر است، که با اعمال قوانین بازی زندگی برای این قالب، بعد از چند مرحله، یک گلایدر تولید می‌کند [۱۴] و با ادامه بازی این روند تکرار می‌شود؛ بنابراین تفنگ می‌تواند رشته‌ای از گلایدرها در شبکه سلولی ساطع کند. تعداد مراحلی که لازم است قوانین بازی اعمال شوند تا یک گلایدر توسط تفنگ تولید شود، دوره تناوب تفنگ نامیده می‌شود [۹]. باتوجه به این‌که در هر دوره تناوب یک گلایدر توسط تفنگ تولید می‌شود و از طرفی گلایدر یک قالب متحرک است، رشته گلایدر ساطع شده رشته‌ای متحرک بوده که فاصله بین تمام گلایدرها در آن یکسان است و این فاصله بستگی به دوره تناوب تفنگ مولد آن دارد. هرچه دوره تناوب تفنگ کوچک‌تر باشد، فاصله‌ی بین گلایدرها در رشته کم‌تر است.

الگوریتم کشف تفنگ مولد یک نوع گلایدر، روشی مبنتی بر الگوریتم ژنتیک است که خارج از بحث این مقاله می‌باشد [۶،۵،۴]. در شکل ۴ تفنگ مربوط به گلایدر شکل ۳ نشان داده شده که دوره تناوب آن برابر ۳۰ است و در ادامه مقاله نیز از آن استفاده می‌شود [۱۷،۲].

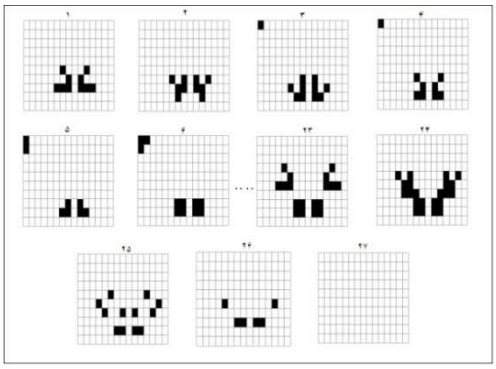
۴-۲- برخورد بین گلایدرها

باتوجه به پیکربندی اولیه‌ای که برای تفنگ گلایدر در نظر گرفته می‌شود، جریان حرکت گلایدرهای تولید شده توسط تفنگ شکل (۴) می‌تواند به سمت راست-پایین یا چپ-پایین شبکه سلولی باشد. با در نظر گرفتن این موضوع می‌توان تقسیم‌بندی تفنگ-راست و تفنگ-چپ را برای یک تفنگ گلایدر عنوان کرد. برای ساده‌سازی این برخورد و کاهش زمان اجرا و همچنین باتوجه به کاربردی که در این مقاله مورد نظر است، می‌توان از شبیه‌سازی برخورد استفاده کرد.

اگر در یک شبکه سلولی دو تفنگ غیر هم‌جهت، به‌طور هم‌زمان گلایدر تولید کنند، امکان برخورد بین گلایدرها در ادامه مسیر آن‌ها وجود دارد و نتیجه برخورد بین دو رشته گلایدر نابودی هر دوی آن‌ها می‌باشد [۸]. در شکل ۵ برخورد بین دو رشته گلایدر نشان داده شده است. همان‌طور که مشخص است، برخورد بین دو گلایدر به تنهایی باعث نابودی کامل آن‌ها نمی‌شود و بعد از اضافه شدن یک گلایدر دیگر از هر رشته به این برخورد و پس از ۲۷ مرحله ادامه بازی، کل سلول‌های مربوط به این چهار گلایدر می‌میرند. بنابراین این نابودی در ۲۷ مرحله و توسط چهار گلایدر رخ می‌دهد [۸،۲].



شکل ۴: تفنگ گلایدر



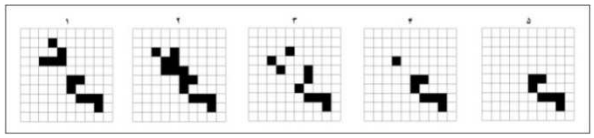
شکل۵: برخورد بین گلایدرها

همان‌طور که در شکل ۲ مشخص است، یکی از قالب‌های ثابت در بازی زندگی، بلاک می‌باشد. همچنین در مرحله ششم از شکل ۵ پدیدار شدن دو بلاک در اثر برخورد بین دو گلایدر دیده می‌شود. در ادامه دو گلایدر به این بلاک‌ها برخورد کرده که باعث از بین رفتن هر چهار گلایدر می‌شود. بنابراین می‌توان نتیجه گرفت که برخورد بین یک گلایدر و یک بلاک باعث از بین رفتن هر دوی آن‌ها می‌شود. از همین ویژگی می‌توان در پیاده‌سازی برخورد بین گلایدرها استفاده کرد. در این حالت زمانی که دو گلایدر ساطع شده هم‌زمان از دو تفنگ غیر هم‌جهت، در یک فاصله نزدیک و زاویه‌ای مناسب از هم قرار داشته باشند، برخورد بین آن‌ها پیش‌بینی می‌شود و می‌توان با قرار دادن دو بلاک در ادامه مسیر آن‌ها به مرحله ۲۳ از شکل

۵ پرش کرد. فاصله و زاویه‌ی دو گلایدر باید مطابق مرحله ۲۳ام از شکل ۵ باشد تا بتوان برخورد را پیش‌بینی کرد. با این شرایط تعداد گلایدرهای مورد نیاز به دو گلایدر و تعداد مراحل جهت کامل شدن برخورد به پنج مرحله کاهش پیدا می‌کند. در ادامه مقاله نیز منظور از برخورد بین گلایدرها، برخورد شبیه-سازی شده آن‌ها خواهد بود.

۴-۳- خورنده

یکی دیگر از قالب‌های بازی زندگی خورنده نامیده می‌شود. ویژگی خورنده به‌این‌صورت است، که در اثر برخورد گلایدر با آن، بعد از چند مرحله گلایدر را از بین می‌برد و خود بدون تغییر باقی می‌ماند [۷]؛ یعنی گلایدر را می‌خورد. شکل ۶ یکی از انواع خورنده‌ها و برخورد آن با گلایدر را نشان می‌دهد. برای پیاده سازی در ادامه مقاله از همین نوع خورنده استفاده شده است.



شکل (۶): خورنده و برخورد گلایدر با آن

۵- پیاده سازی گیت‌های منطقی

یک گیت منطقی عملگری دودویی بوده و دارای ورودی و خروجی دودویی است. از مهم‌ترین گیت‌های منطقی می‌توان به گیت AND، OR وNOT اشاره کرد که در جدول شماره ۱ ورودی و خروجی آن‌ها مشخص شده است؛ A و B ورودی‌های گیت را تعیین می‌کنند.

جدول ۱: گیت‌های AND,OR,NOT

گیت	B	A	خروجی
AND	0	0	0
	0	1	0
	1	0	0
	1	1	1
OR	0	0	0
	0	1	1
	1	0	1
	1	1	1
NOT	1		0
	0		1

همان‌طور که مشخص است، در پیاده‌سازی گیت‌های منطقی سه نکته مهم وجود دارد: پالس‌های ورودی و خروجی گیت، سیم‌های حامل پالس‌های الکتریکی و بخش پردازش جهت محاسبه و تعیین پالس خروجی. اکنون برای پیاده‌سازی این گیت‌ها به کمک گلایدرها می‌بایست این سه نکته اصلی را به مفاهیم مربوط به گلایدرها ارتباط داد. برای این منظور می‌توان این نگاشت را در نظر گرفت: پالس‌های الکتریکی: رشته گلایدرها، سیم‌های حامل: مسیر حرکت یک رشته از گلایدرها، بخش پردازش: برخورد بین گلایدرها [۲].

با درنظر گرفتن نگاشت بالا، به‌ازای هر پالس ورودی گیت، یک تفنگ گلایدر استفاده می‌شود تا رشته گلایدرهای ساطع شده از آن، مشخص کننده آن پالس ورودی باشد. همچنین مسیر حرکت این رشته از گلایدرها معرف همان سیم حامل پالس‌های الکتریکی خواهد بود. برای تعیین خروجی گیت از برخورد بین گلایدرها و نتایج حاصل از آن استفاده شده و با توجه به نوع گیت، تعدادی تفنگ گلایدر برای پیاده‌سازی برخوردها در نظر گرفته می‌شود. این تفنگ‌ها که نقشی در تولید ورودی گیت ندارند و یک رشته گلایدر ثابت برای پیاده‌سازی برخوردهای لازم، جهت تعیین خروجی تولید می‌کنند، تفنگ پردازش نامیده می‌شوند. همچنین برای نابود کردن برخی از گلایدرها که نقشی در تعیین نتیجه ندارند، از خورنده استفاده می‌شود. با این توضیحات آن‌چه برای پیاده‌سازی سه گیت اصلی درفضای اتوماتای سلولی و برمبنای قوانین بازی زندگی لازم خواهد بود، در جدول شماره ۲ مشخص شده است.

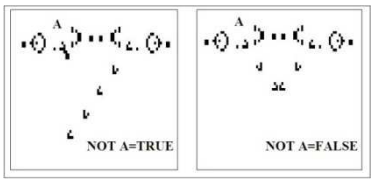
جدول ۲: نیازهای پیاده‌سازی سه گیت اصلی

گیت	تعداد تفنگ های ورودی	تعداد تفنگ های پردازش	تعداد برخوردهای تاثیرگذار	تعداد خورنده‌ها
AND	۲	۱	۲	۱
OR	۲	۲	۳	۱
NOT	۱	۱	۱	۰

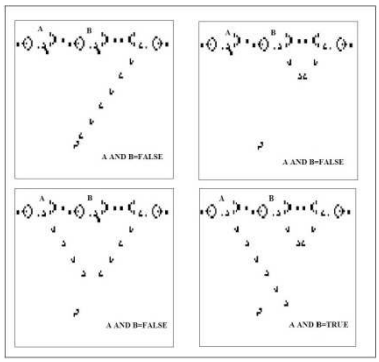
در هر دوره تناوب یک‌بار به گیت، ورودی اعمال می‌شود و این درست در مرحله‌ای است که تفنگ‌ها گلایدر تولید می‌کنند. زمانی که ورودی گیت صفر است (false)، از یک بلاک در نزدیک‌ترین وضعیت به تفنگ، برای از بین بردن گلایدر ساطع شده استفاده می‌شود؛ به‌این‌شکل ورودی صفر برای گیت پیاده‌سازی می‌شود و هرگاه ورودی یک (true) باشد؛ بدون درنظر گرفتن بلاک، اجازه پیش‌روی گلایدر به شبکه سلولی داده می‌شود. به این ترتیب رشته‌ای از گلایدرها معادل رشته‌ای از صفرها و یک‌ها خواهد بود؛ به‌طوری‌که هر گلایدر معرف عدد یک و هر جای خالی گلایدر معرف عدد صفر است.

برای مشخص کردن خروجی، یک مختصات مشخص در اتوماتا درنظرگرفته می‌شود؛ به‌طوری‌که پایین‌تر از آخرین برخورد ممکن بوده و همچنین در امتداد مسیر گلایدرهایی باشد که

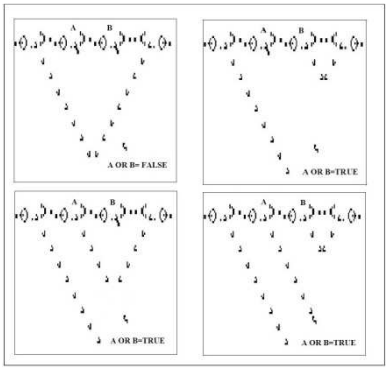
نقش گلایدر خروجی را دارند. بعد از گذشت تعداد مراحل لازم برای رسیدن اولین گلایدر به این مختصات، این مختصات در بازه‌های زمانی به فاصله ۳۰(دوره تناوب تفنگ) بررسی می‌شود. اگر گلایدری در آن مختصات وجود داشته باشد، خروجی گیت در آن لحظه یک و در غیر این صورت صفر است. باتوجه به توضیحات ارائه شده، پیاده‌سازی سه گیت اصلی در شکل‌های ۶، ۷ و ۸ آورده شده است.



شکل۶: پیاده‌سازی گیت NOT



شکل ۷: پیاده‌سازی گیت AND



شکل۸: پیاده‌سازی گیت OR

۶- پیاده‌سازی سایر توابع منطقی

با وجود سه گیت اصلی مطرح شده، سایر گیت‌ها و توابع منطقی به صورت ترکیبی از این سه گیت قابل پیاده‌سازی هستند. تنها نکته‌ای که در این پیاده‌سازی‌ها باید مورد توجه قرار گیرد، تعداد تفنگ‌های پردازش است. زیرا هرچه تعداد این تفنگ‌ها کم‌تر باشد، تعداد برخوردهای لازم جهت تعیین خروجی کم‌تر بوده و خروجی آسان‌تر و سریع‌تر به‌دست خواهد آمد. بنابراین باید تابع منطقی مورد نظر با استفاده از سه گیت اصلی به شکلی پیاده‌سازی شود، که کم‌ترین تفنگ لازم باشد.

۶-۱ پیاده‌سازی گیت XOR

با توجه به رفتار گیت XOR که درجدول شماره ۳ مشخص شده است، چندین تابع منطقی به صورت ترکیبی، برای تعریف این گیت وجود دارد. اما برای پیاده‌سازی آن در محیط اتوماتای سلولی و به کمک گلایدرها باید به دنبال تعریفی بود که کم‌ترین تفنگ را لازم داشته باشد. برای مثال به مقایسه دو عبارت (۲) و (۳) برای این گیت پرداخته می‌شود [۲].

$$A \text{ XOR } B = (A \text{ AND } (\text{NOT } B)) \text{ OR } ((\text{NOT } A) \text{ AND } B) \quad (۲)$$

$$A \text{ XOR } B = \text{NOT}((A \text{ AND } B)) \text{ AND } (A \text{ OR } B) \quad (۳)$$

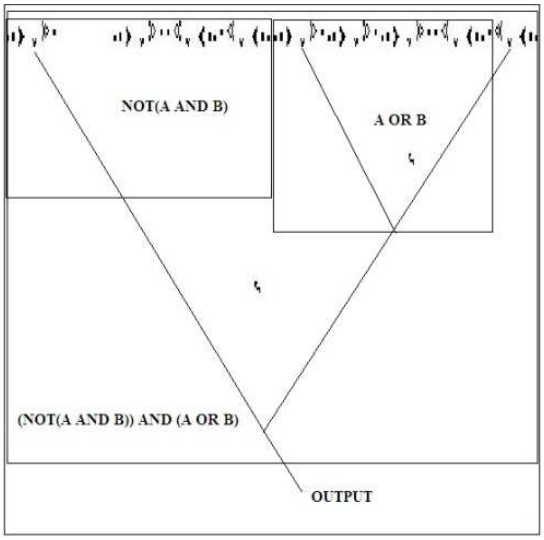
جدول۳: عملکرد گیت XOR

A	B	خروجی
0	0	0
0	1	1
1	0	1
1	1	0

در عبارت (۲) با توجه به اولویت عملگرها ابتدا باید (A AND (NOT (B)) پیاده‌سازی شود که با توجه به جدول شماره (۲)، دو تفنگ پردازش و دو تفنگ ورودی نیاز است. همچنین برای پیاده‌سازی عبارت (NOT A) AND B دو تفنگ پردازش و دو تفنگ ورودی نیاز می‌باشد. اکنون خروجی این دو عبارت به عنوان ورودی‌های گیت OR میانی درنظر گرفته شده که این گیت به دو تفنگ پردازش نیاز دارد. بنابراین به‌طور کلی برای پیاده‌سازی عبارت (۲) ده تفنگ لازم است. اما در عبارت (۳) پیاده‌سازی عبارت NOT (A AND B)، دو تفنگ پردازش و دو تفنگ ورودی و پیاده‌سازی عبارت A OR B، دو تفنگ پردازش و دو تفنگ ورودی نیاز دارد. و در آخر با درنظر گرفتن خروجی این دو عبارت به عنوان ورودی گیت AND میانی، به یک تفنگ پردازش دیگر نیاز خواهد بود. بنابراین به طورکلی پیاده‌سازی عبارت (۳)، نه تفنگ نیاز خواهد داشت. با این بررسی می‌توان نتیجه گرفت که عبارت (۳) برای پیاده‌سازی گیت XOR مناسب‌تر است. در شکل ۹ این پیاده سازی نشان داده شده است.

باتوجه به فرد بودن تعداد تفنگ‌ها و از آن‌جا که گلایدرهای خروجی در راستای مسیر اولین تفنگ از سمت چپ هستند، باید در طراحی ایده‌ای در نظر گرفته شود تا دو گلایدر که همزمان ساطع می‌شوند و به‌هم برخورد می‌کنند، مسیری با طول

یکسان طی کرده باشند. دراین صورت خروجی درستی تولید می‌شود. برای این منظور همان‌طور که در شکل ۹ دیده می‌شود، فضایی خالی به اندازه‌ی یک تفنگ، بین دو تفنگ اول از سمت چپ درنظر گرفته شده است.



شکل ۹: پیاده‌سازی گیت XOR

۷- رمزنگاری متن

در بخش‌های قبل مفاهیم گلایدرها و پیاده‌سازی توابع منطقی به کمک آن‌ها بررسی شد. در این بخش به عنوان یکی از کاربردهای ممکن به بررسی رمزنگاری متن پرداخته می‌شود.

۷-۱- تبدیل متن به گلایدر

اولین نکته‌ای که در این پیاده‌سازی به نظر می‌رسد، تبدیل متن به رشته‌ای از گلایدر است. با درنظر گرفتن کد اسکی کاراکترها و توجه به منحصر بودن این کد برای تمام کاراکترها، می‌توان به هر کاراکتر یک رشته دودویی به طول هفت بیت اختصاص داد که همان نمایش دودویی کد اسکی مربوط به کاراکتر خواهد بود. اکنون باید این رشته دودویی به رشته گلایدر تبدیل شود. با درنظر گرفتن دو قانون زیر این تبدیل به‌صورت یک‌به‌یک امکان پذیر است. (۱): عدد یک در رشته دودویی نشان دهنده وجود گلایدر است. (۲): عدد صفر در رشته دودویی نشان دهنده عدم وجود گلایدر است.

در این شرایط می‌توان متن را به رشته‌ای از گلایدرها تبدیل کرد. به این ترتیب که ابتدا کد اسکی متن به‌صورت دودویی تولید شده و سپس در این کد به‌ازای هر بیت با مقدار یک، گلایدر و به هر بیت با مقدار صفر یک جای خالی گلایدر نگاشت داده می‌شود. نمونه‌ای از این تبدیل برای عبارت glider در جدول شماره ۴ آمده است.

جدول۴ : مثال تبدیل متن به رشته گلایدر

کاراکتر	کد اسکی	کد اسکی دودویی
G	67	1000011
L	108	1101100
I	105	1101001
D	64	1000000
E	65	1000001
R	72	1001000

کد دودویی معادل با متن اصلی:

10000111101100110100110000001000000110010000

اکنون برای تولید رشته گلایدر مربوط به متن باید از یک تفنگ گلایدر استفاده کرد و در بازه‌هایی زمانی به فاصله ۳۰ که برابر با دوره تناوب تفنگ گلایدر است، به کنترل گلایدرهای تولید شده پرداخت. در هر دوره تناوب هرگاه در کد دودویی، عدد یک وجود داشته باشد گلایدر تولید شده در آن بازه توسط تفنگ، نشان دهنده آن است و هرگاه صفر وجود داشته باشد، استفاده از یک بلاک باعث از بین رفتن گلایدر در آن بازه شده و جای خالی آن در رشته نشان دهنده عدد صفر خواهد بود.

قابل مشاهده است که در این تبدیل طول رشته گلایدر هفت برابر طول متن اولیه است، زیرا هر کاراکتر به یک رشته گلایدر به طول هفت تبدیل می‌شود. می‌توان با در نظر گرفتن تعداد انواع کاراکترهای موجود در متن اصلی، طول رشته گلایدرها را کاهش داد. به‌این معنی که در این شرایط می‌توان یک جدول مشابه جدول کد اسکی برای کاراکترها ایجاد کرد و در آن یک رابطه یک‌به‌یک بین کد و کاراکتر تولید کرد. دراین‌صورت طول رشته دودویی و به دنبال آن طول رشته گلایدر مربوط به هر کاراکتر کاهش می‌یابد و در نهایت طول رشته گلایدر مربوط به متن اصلی کاهش پیدا خواهد کرد. روشن است که با این کاهش، زمان لازم برای رمز کردن متن، نیز کم می‌شود.

۷-۲- انتخاب کلید رمز

نکته دوم انتخاب کلید مناسب برای رمز کردن متن است. این کلید می‌تواند به‌صورت تصادفی توسط یک مولد اعداد تصادفی و متناسب با طول متن اصلی تولید شود. بعد از تولید کلید تصادفی، رشته دودویی معادل با آن مطابق با روش مطرح شده در بخش قبل، به رشته گلایدر تبدیل می‌شود. همچنین می‌توان کلید را به صورت یک رشته از کاراکترها در نظر گرفت که این رشته در نهایت باید به کد دودویی معادل با آن و سپس رشته گلایدر معادل تبدیل شود. برای انجام این تبدیلات مراحل

مشابه با تبدیل متن اصلی به رشته دودویی و رشته گلایدر مربوط به آن می‌باشد. اگر طول رشته دودویی کلید کم‌تر از طول رشته دودویی متن باشد، باید از تکرار متوالی کلید به رشته‌ای دودویی رسید که طولی برابر با طول رشته دودویی معادل با متن اصلی داشته باشد و از این رشته به عنوان کلید دودویی رمز استفاده کرد. قابل ذکر است که در این رشته ممکن است آخرین تکرار به صورت کامل نباشد که مشکلی در ادامه مراحل به وجود نخواهد آورد.

۷-۳- عملیات رمزنگاری

دو رشته گلایدر تولید شده در بخش‌های ۷-۲ و ۷-۳ به عنوان ورودی‌های گیت XOR طراحی شده در شکل ۹ استفاده می‌شوند و خروجی به صورت رشته‌ای از گلایدرها در اتنها به‌وجود خواهد آمد که این رشته با اجرای عکس عملیات تبدیل متن به گلایدر، قابل تبدیل به متن خواهد بود.

برای رمزگشایی متن نیز می‌توان متن رمز شده را مجددا با همان کلید، به گیت اعمال کرد و رشته گلایدر معادل متن اصلی را به‌دست آورد. بنابراین مشاهده می‌شود که روش رمزنگاری ارائه شده یک روش متقارن می‌باشد زیرا در هر دو مرحله رمزنگاری و رمزگشایی از یک کلید یکسان استفاده می‌شود. در این روش تابع رمز عملگر XOR می‌باشد که در شکل ۹ نشان داده شده است.

۸- نتیجه گیری

در این مقاله با بررسی ویژگی‌های گلایدرها در بازی زندگی، نشان داده شد که چگونه می‌توان به شبیه‌سازی رفتار گیت‌های منطقی پایه در اتوماتای سلولی دست یافت. همچنین می‌توان از ترکیب این گیت‌های پایه به پیاده‌سازی سایر گیت‌ها و توابع منطقی به در فضای اتوماتای سلولی رسید. با بسط این موضوع به ورودی‌های چند بیتی، امکان پیاده‌سازی سایر محاسبات دودویی در اتوماتای سلولی فراهم می‌شود. در ادامه با کنترل گلایدرهای ساطع شده از تفنگ گلایدر به‌وسیله قالب خورنده و بلاک، امکان نگاشت متن به رشته‌ای از گلایدرها نشان داده شد. به‌این‌ترتیب امکان استفاده از گلایدرها در مباحث غیر دودویی نیز مشخص می‌شود.

یک رشته از گلایدرها می‌تواند حاوی یک پیام باشد و با برقراری ارتباط بین مفاهیم رمزنگاری و مفاهیم گلایدرها می‌توان به رمزنگاری این پیام در اتوماتای سلولی دست یافت. برای این منظور از توابع و گیت‌های شبیه‌سازی شده به‌عنوان تابع رمز استفاده شده و کلید رمز نیز به صورت رشته‌ای از گلایدرها تعریف می‌شود. خروجی رمز نیز به صورت رشته‌ای از گلایدرها خواهد بود که حاوی متن رمز شده می‌باشد. عملیات رمزگشایی نیز به‌طور مشابه مبتنی بر تئوری گلایدرها انجام می‌گیرد. به‌این‌ترتیب می‌توان یک روش رمزنگاری متقارن مبتنی بر گلایدرها در تئوری بازی زندگی معرفی کرد. پیشنهادی که در ادامه کار می‌تواند موثر باشد بسط این موضوع به فضای موازی است که در آن صورت می‌توان نتایج رمزنگاری را در زمینه‌های مختلفی از جمله زمان اجرا ارتقا داد.

مراجع

[۱] رضا رستگار، محمد رضا میبدی/ "مدل یادگیری Q سلولی و کاربردهای آن"/ دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر

[2] Jean-Phiippe Rannard, "*Implementation of logical functions in the Game of Life*"In A. Adamatzky(Ed), collision-based computing, pp. 491-512, London, 2002.

[3] Emannuel Sapin, L.Ball, A.Adamatzky, "*Genetic approaches to search for computing pattern in cellular automata*", IEEE computational intelligence magazine, university of England, UK, August 2009.

[4] Emmanuel Sapin, Larry Bull, "*Evolutionary search for cellular automata logic gates with collision based computing*", Faculty of computing, engineering and mathematical science, university of the west of England, complex systems, 2008.

[5] Emmanuel sapin, Olivier Bailleux, Jean-Jacques Chabrier, "*Research of a cellular automaton simulating logic gates by evolutionary algorithms*", university Bourgogne, LERSIA, 2008.

[6] Emmanuel Sapin, Olivier Bailleux, Jean-Jacques Chabrier, "*Research of complex form in the cellular automata by evolutionary algorithms*", P.liardet et al.(Eds.): EA 2003, LNCS 2936, pp. 357-367, University Bourgogne, 2004.

[7] E.spain, O.Bailleux, J-J.Chabrier, P.Collet, "*Demonstration of the universality of a new cellular automata*", International Journal of Unconventioned Computing, vol.3, pp.79-103, University Bourgogne, 2007.

[8] Emmanuel sapin, Olivier Bailleux, Jean-Jacques Chabrier , "*A new approach of stream duplication in 2D cellular automata*", In GECCO04, Lecture Notes in Computer Science, 3102, pp. 175-187, university Bourgogne , 2004.

[9] E.spain, L.Bull, "*Searching for glider guns in cellular automata: exploring evolutionary and other techniques*", Lecture Notes in computer Science, university of the west of England, 2008.

[10] S.Wolfram, "*Universality and complexity in cellular automata*", In Physica D: Nonlinear Phenomena,vol. 10, pp.1-35, 1984.

[11] Kumar, Tapas, Sahoo, G, "*A Novel Method of Edge Detection using Cellular Automata*", International Journal of Computer Application, Vol.9-No.4, pp. 38-44, November 2010.

[12] Gardner, Martin, Mathematical Games, The fantastic combinations of John Conway's new solitaire game life, pp. 120–123. ISBN 0894540017. Archived from the original on 2009-06-03.

[13] Paul Chapman, Life Universal Computer, Retrieved July12,2009,http://www.igblan.freeonline.co.uk/ igblan/ca/

[14] E.spain, L.Bull ," the emergence of glider guns in cellular automata found by evolutionary algorithms", , engineering and mathematical science, university of the west of England ,2008.

[15] Stephen A. Silver, Glider, The Life Lexicon, Retrieved July 12, 2009.

[16] Jason Summers, Game of Life Status page, retrieved 2012-02-23.

[17] Stephen A. Silver, Gosper glider gun, The Life Lexicon.

[18] Andrzej Okrasinski, Game of Life Object Statistics, Archived from the original on 2009-07-27.