يك الگوريتم مرتب‌سازي موازي براي اتوماتاي سلولي خطي

سيد ميثم حسيني سدهي   محمد رضا ميبدي
دانشكده مهندسي كامپيوتر و فناوري اطلاعات
دانشگاه صنعتي اميركبير
تهران   ايران
mmeybodi@aut.ac.ir , sehrestan@Hotmail.com

چكيده

مرتب‌سازي يكي از مسائل كاربردي در بسياري از علوم مهندسي مي‌باشد. به همين جهت الگوريتم‌هاي متنوعي از مرتب‌سازي با ديدگاه‌هاي متفاوت و پيچيدگي‌هاي مختلف گزارش شده است. در اين مقاله دو الگوريتم مرتب‌سازي جديد براي اتوماتاي سلولي خطي پيشنهاد شده است. اين دو الگوريتم در مقايسه با تنها الگوريتم گزارش شده براي اتوماتاي سلولي خطي از نوع همسايگي، شعاع همسايگي و قوانين متفاوتي برخوردار است، همچنين ساختار اتوماتاي سلولي استفاده شده در آن نيازي به حافظه اضافي براي مرتب‌سازي ندارد. الگوريتم‌هاي معرفي‌شده در مقايسه با تنها الگوريتم گزارش شده از همسايگي كوچكتري برخوردار مي‌باشند و نيازمند محاسبات كمتري براي مرتب‌سازي مي‌باشند.

كلمات كليدي: الگوريتمهاي مرتب‌سازي، پردازش موازي، اتوماتاي سلولي

Parallel Sorting Algorithms for One-dimensional Cellular Automata

M. Hosseini Sedehi   M. R. Meybodi
Computer Engineering and Information Technology Department
Amirkabir University of Technology
Tehran Iran
mmeybodi@aut.ac.ir , sehrestan@Hotmail.com

Abstract

Sorting is on of the major tasks in engineering science and for this reason variety of sorting algorithms for different computational models with different time complexities are reported in the literatures. In this paper we propose two sorting algorithms for one dimensional cellular automata. The proposed algorithms have lower radius and lower time complexity comparing to the only reported sorting algorithm.

Keywords: Sorting, Cellular Automata, Parallel Computation

۱- مقدمه

اتوماتاي سلولي، مدلي مبتني بر ساختار سلولي بوده كه رفتارش بر اساس ارتباط محلي استوار مي‌باشد. هر سلول بر اساس اطلاعات محلي، اطلاعات موجود در خود را پردازش مي‌كند. قوانين محلي حاكم بر اتوماتاي سلولي در هر مرحله، وضعيت جديد هر سلول را با توجه به وضعيت همسايه‌هاي مجاور سلول بدست مي آورد [1]. تنوع و سادگي كاركرد اتوماتاي سلولي، امكان استفاده در علوم مختلفي مانند، فيزيك، رياضي، كامپيوتر و حتي بيولوژيك را فراهم ساخته است. ارتباط محلي از طريق تعامل همسايگي، پردازش محلي اطلاعات از طريق قوانين محلي حاكم، پردازش موازي اطلاعات در تمامي سلول‌ها به صورت همزمان و توزيعي بودن پردازش اطلاعات موجب گسترش اتوماتاهاي سلولي در زمينه‌هاي مختلف كاربردي، شبيه‌سازي و تحقيقاتي شده است. يكي از موارد مورد توجه محققين، حل مسائل پايه مانند مسائل مرتبط به گراف، مرتب‌سازي اعداد و توليد زنجيره اعداد تصادفي توسط اتوماتاي سلولي مي‌باشد.

مرتب‌سازي داده‌ها يكي از مسائل مورد توجه در كاربردهاي متنوعي چون پردازش تصاوير، محاسبات گراف‌ها، مسائل پايگاه داده،

کاربردهاي اندازه‌گيري و از اين دست مي‌باشد. مي‌توان الگوريتم‌هاي مرتب‌سازي را به دوسته الگوريتم‌هاي ترتيبي و الگوريتم‌هاي موازي تقسيم نمود. الگوريتم‌هاي موازي از ديدگاه نحوه حل مسئله و موارد مورد استفاده آن به دو دسته الگوريتم‌هاي بر مبناي مدل‌هاي رياضي و الگوريتم‌هاي وابسته به بستر پياده‌سازي تقسيم مي‌شوند. الگوريتم‌هاي بر مبناي ماشين‌هاي SIMD از دسته الگوريتم‌هاي وابسته به بستر پياده‌سازي مي‌باشند. در اين الگوريتم‌ها عموماً آرايه يك بعدي از پردازنده‌ها از طريق الگوريتم انتقال زوج و فرد١، n عنصر داده‌اي را با استفاده از n پردازنده مقايسه و مرتب مي‌نمايند. دسته‌اي ديگر از الگوريتم‌هاي وابسته به بستر بر اساس ماشين‌هاي موازي MIMD و بر مبناي الگوريتم مرتب‌سازي سريع٢ مي‌باشند. بسياري از الگوريتم‌هاي ترتيبي مانند مرتب‌سازي انتخابي٣، مرتب‌سازي تركيبي٤ و يا مرتب‌سازي سريع با راهكارهاي متناظر با پردازش موازي، در سيستم‌هاي موازي مورد استفاده قرار گرفته‌اند [3-9].

دسته‌اي ديگر از الگوريتم‌هاي مرتب‌سازي، الگوريتم‌هاي بر مبناي مدل‌هاي رياضي مانند اتوماتاهاي سلولي٥ و آرايه‌هاي تپنده٦ [2] بوده كه منفك شده از مفهوم بستر پياده‌سازي مي‌باشند. اگرچه الگوريتم‌هاي بر مبناي مدل‌هاي رياضي از ديدگاه‌هاي مختلف پردازش موازي حاصل شده‌اند، با اين وجود بر اساس قوانين حاكم بر محيطشان، قابليت تبديل شدن به يكديگر را دارا مي‌باشند.

در الگوريتم‌هاي مبتني بر اتوماتاي سلولي خطي، محيط حاكم بر مرتب‌سازي، سلول‌هاي اتوماتاي سلولي، همسايگي‌هاي سلولي و قوانين محلي حاكم بر سلول‌هاي اتوماتاي سلولي مي‌باشند. بر روي مرتب‌سازي از ديدگاه اتوماتاي سلولي خطي، كار زيادي صورت نگرفته و معروفترين مرتب‌سازي بر پايه اتوماتاي سلولي توسط گورديلو و لونا پيشنهاد شده است[10].

در الگوريتم اول گورديلو_لونا تعداد همسايگان مورد بررسي هر سلول سه و در الگوريتم دوم گورديلو_لونا تعداد همسايگان مورد بررسي هر سلول، پنج سلول مي‌باشد. براي مرتب‌سازي آرايه‌ها در الگوريتم گورديلو_لونا هر سلول نيازمند سه حافظه خواندني و نوشتني بوده كه يكي از آنها در بردارنده مقدار عددي و دو حافظه ديگر شامل كنترل كننده‌هاي جابجايي مقدار عددي به سمت چپ و راست مي‌باشند. اين كنترل‌كننده ها علاوه بر تعيين جابجايي براي عدم تصادم مقادير سلول‌ها در نظر گرفته شده است. در اين الگوريتم‌ها در هر مرحله، محتواي دو سلول اتوماتا با يكديگر بر اساس قانون حاكم تعويض مي‌شوند. همچنين بر اساس ساختار جابجايي اين الگوريتم‌ها، در يک مرحله، جابجايي محتواي يک سلول به سمت چپ و راست بررسي شده و در مرحله دوم، محتواي دو سلول همسايه بر اساس بررسي انجام شده و مقدار دو حافظه كنترل كننده جابجايي، جابه‌جا مي‌شوند.

در اين مقاله دو الگوريتم مرتب‌سازي براي اتوماتاي سلولي خطي پيشنهاد شده است. هر سلول در الگوريتم پيشنهادي اول داراي دو همسايه سمت راست بوده و در الگوريتم پيشنهادي دوم تعداد همسايگان هر سلول برابر چهار مي‌باشد. الگوريتم دوم با دو نوع همسايگي متقارن و همسايگي نامتقارن پياده‌سازي شده است. دو الگوريتم پيشنهادي، پيچيدگي زماني مشابهي با الگوريتم‌هاي گورديلو_لونا دارند، ولي از نوع همسايگي، شعاع همسايگي و قوانين متفاوتي برخوردار مي‌باشند. همچنين در ساختار استفاده شده براي مرتب‌سازي نيازي به حافظه اضافي براي كنترل جابه‌جايي مقادير سلول‌هاي همسايه وجود ندارد. الگوريتم‌هاي پيشنهادي در مقايسه با الگوريتم‌هاي گورديلو_لونا از همسايگي كوچكتري برخوردار مي‌باشند. در الگوريتم‌هاي پشنهادي بر خلاف الگوريتم‌هاي گورديلو_لونا، بررسي جابجايي و جابجايي مقادير سلول‌هاي همسايه بر اساس قانون حاكم بر اتوماتاي سلولي تنها در يک مرحله صورت گرفته و نيازمند دو مرحله مجزا نمي‌باشد. حذف يک مرحله اجرايي موجب افزايش سرعت الگوريتم مرتب‌سازي مي‌شود.

ادامه مقاله بدين صورت سازماندهي شده است. در بخش ۲ اتوماتاي سلولي به اختصار شرح داده ميشود. در بخش ۳ ابتدا الگوريتم مرتب‌سازي گورديلو_لونا و سپس دو الگوريتم پيشنهادي معرفي مي‌شوند. بخش ۴ نتيجه گيري ميباشد.

۲- اتوماتاي سلولي

اتوماتاي سلولي مدلي رياضي است كه در آن چندين مؤلفه ساده براساس يك رابطه معين با يكديگر همكاري نموده و توانايي ايجاد الگوهاي پيچيده را دارند. اتوماتاي سلولي از يك شبكه منظم سلولي تشكيل شده كه در آن هر سلول مي‌تواند $r > 1$ مقدار مختلف اختيار نمايد. سلول‌هاي اتوماتاي سلولي در زمان‌هاي گسسته و بطور همزمان و بر طبق يك قانون محلي بنام Φ بهنگام مي‌شوند كه در آن مقدار هر سلول براساس مقادير سلول‌هاي همسايه تعيين مي‌شود.

اتوماتاي سلولي براساس معيارهاي مورد بررسي به دسته‌هاي مختلف تقسيم مي‌گردد. براساس معيار بعد شبكه، اتوماتاي سلولي به اتوماتاي سلولي يك بعدي، دوبعدي و غيره تقسيم شده و براساس مقدار r به اتوماتاي سلولي دودوئي $(r = 1)$ و اتوماتاي سلولي چند مقداره تقسيم مي‌شود [1]. اتوماتاي سلولي d بعدي يك چندتايي $CA = (Z^d, \phi, N, \Phi)$ است به طوريكه:

- Z^d يك شبكه از d-تايي‌هاي مرتب از اعداد صحيح مي‌باشد. اين شبكه مي‌تواند يك شبكه متناهي، نيمه متناهي يا نامتناهي باشد.
- $\phi = \{1, \cdots, m\}$ يك مجموعه متناهي از حالت‌ها است.
- $N = \{\bar{x}_1, \cdots, \bar{x}_m\}$، $\bar{x}_i \in Z^d$، يك زير مجموعه متناهي از Z^d بوده كه بردار همسايگي خوانده مي‌شود. بردار همسايگي، موقعيت نسبي همسايگان براي هر سلول u در شبكه سلولي توسط رابطه (١) مشخص مي‌شود.

$$N(u) = \{u + \bar{x}_i \mid i = 1,\dots,\overline{m}\} \qquad (1)$$

تابع $N(u)$ دو شرط زير را ارضا مي‌كند:

$$\forall u \in Z^d \Rightarrow u \in N(u) \qquad (2)$$
$$\forall u, v \in Z^d \Rightarrow u \in N(v) \wedge v \in N(u)$$

- $\Phi : \underline{\phi}^{\overline{m}} \to \underline{\phi}$ قانون محلي اتوماتاي سلولي بوده كه به سه دسته قانون عام[7]، كلي‌گرا[8] و كلي‌گرا بيروني[9] تقسيم مي‌شود.

در اتوماتاي سلولي يك بعدي مقدار سلول i (براي $1 \le i \le n$) در زمان t كه با $a_i(t)$ نشان داده مي‌شود، طبق رابطه (۳) محاسبه مي‌گردد:

$$a_i(t+1) = \Phi[a_{i-1}(t), a_i(t), a_{i+1}(t)] \qquad (۳)$$

در رابطه فوق، اگرقانون Φ فقط به مقدار همسايه‌ها بستگي داشته باشد آنرا قانون عام و اگر قانون Φ تابعي از مجموع مقادير سلول‌هاي همسايه يك سلول مركزي باشد آنرا قانون كلي‌گرا مي‌نامند و طبق رابطه (۴) بيان مي‌شود :

$$a_i(t+1) = \Phi[a_{i-1}(t) + a_i(t) + a_{i+1}(t)] \qquad (۴)$$

در صورتي كه قانون تابعي از مجموع مقادير سلول‌هاي همسايه و مقدار سلول مركزي باشد آنرا قانون كلي‌گراي بيروني مي‌گويند و بصورت رابطه (۵) نشان داده مي‌شود.

$$a_i(t+1) = \Phi[a_i(t), a_{i-1}(t) + a_{i+1}(t)] \qquad (۵)$$

بر اساس تعاريف فوق اتوماتاي سلولي را مي‌توان، سيستم محاسباتي توزيع شده‌اي دانست كه از عناصر پردازشي همسان بسيار ساده‌اي شكل گرفته كه از تعامل بين اين عناصر ساده، بروز رفتارهاي پيچيده امكان‌پذير مي‌باشد. در اين سيستم محاسباتي، قوانين محاسباتي بر اساس همسايگي و به صورت محلي تعريف مي‌شوند.

در اين مقاله به اتوماتاي سلولي از ديد يك ماشين محاسباتي عمومي نگريسته شده كه براي حل مسئله مرتب‌سازي مورد استفاده قرار گرفته است. براي حل مسائل محاسباتي نياز به ساختمان داده‌اي شامل ورودي، خروجي و يك رويه براي تبديل ورودي به خروجي مي‌باشد. مراحل پردازش و تبديل ورودي به خروجي در اتوماتاي سلولي توسط انتقال حالت‌هاي اتوماتاي سلولي پياده سازي مي‌شود. هرچند در تعريف اتوماتاي سلولي استاندارد حافظه منظور نشده است، ولي اگر اتوماتاي سلولي بخواهد نقش يك ماشين محاسبه‌گر عمومي را بازي كند اولاً هر سلول نياز به تعدادي حافظه داشته و همچنين بايستي قابليت نوشتن مقادير خروجي در حافظه را داشته باشد.

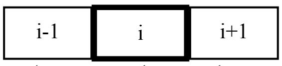
۳- الگوريتم‌هاي مرتب‌سازي اعداد با اتوماتاي سلولي خطي

در اين بخش ابتدا الگوريتم مرتب‌سازي گورديلو_لونا شرح داده خواهد شد، سپس دو الگوريتم پيشنهادي معرفي مي‌شوند.

۳-۱- الگوريتم مرتب سازي گورديلو_لونا

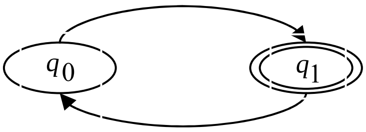
در الگوريتم‌هاي گورديلو_لونا با تركيب اتوماتاي سلولي با اتوماتاي ميلي، ماشين محاسبه‌گري ايجاد شده كه هر سلول آن مجهز به سه خانه حافظه بوده و همچنين قابليت نوشتن مقادير خروجي در حافظه امكان پذير مي‌باشد.

همسايگي استفاده شده در الگوريتم مرتب‌سازي اول، از نوع متقارن با شعاع يک همانند شکل (۱) مي‌باشد. در اين الگوريتم، مرتب‌سازي داراي دو مرحله q_0 و q_1 همانند شكل (۲) بوده كه در ابتدا تمامي سلول‌هاي اتوماتاي سلولي در مرحله q_0 قرار دارند.



شكل(۱): همسايگي در الگوريتم اول گورديلو_لونا.

در انتقال از مرحله q_0 به مرحله q_1، مقدار هر سلول با مقدار دو سلول همسايه‌اش مقايسه شده و در صورتي كه مقدار سلول مركزي از مقدار سلول همسايه سمت راستش بزرگتر باشد، آن را به سمت راست خود و در صورتي كه مقدارش از مقدار سلول همسايه سمت چپش كوچكتر باشد، آن را به سمت چپ خود انتقال مي‌دهد. بنابر اين يك سلول مي‌تواند، در يك زمان مقدارش را هم با همسايه سمت راست و هم با همسايه سمت چپش جابجا نمايد، كه در اين صورت تصادم رخ مي‌دهد.



شكل(۲): مراحل اجرايي الگوريتم گورديلو_لونا

با اعمال كنترل در جابجايي، از بروز تصادم مي‌توان جلوگيري نمود. براي اين منظور در هر سلول دو حافظه با نام‌هاي سد كننده راست و چپ در نظر گرفته مي‌شود. با اين روش، هر سلولي كه مقدارش كوچكتر يا مساوي مقدار سلول همسايه سمت راستش باشد، مقدار سد كننده سمت راست خود را برابر يك و در صورتي كه مقدارش كوچكتر از مقدار همسايه سمت چپش و كوچكتر از همسايه راستش باشد، مقدار سد كننده چپ خود را برابر يك قرار مي‌دهد. مقادير سد كننده راست و چپ بر پايه روابط (۶) و (۷) محاسبه مي‌شوند.

$$S_i^r = \begin{cases} 1 & x_i > x_{i+1} \\ 0 & Otherswise \end{cases} \qquad (۶)$$

$$S_i^l = \begin{cases} 1 & x_i < x_{i-1} \wedge x_i \le x_{i+1} \\ 0 & Otherswise \end{cases} \qquad (۷)$$

در رابطه فوق S_i^r سد كننده راست و S_i^l سد كننده چپ و x_i بيانگر مقدار سلول مركزي i-ام مي‌باشند.

تصميم‌گيري براي جابجايي بر اساس مقدار سد كننده چپ و راست انجام مي‌شود. اگر سد كننده چپ سلول و سد كننده راست سلول همسايه سمت چپ سلول برابر يك باشد، مقدار همسايه چپ سلول جايگزين مقدار اين سلول مي‌شود و در صورتي كه مقدار سد كننده راست سلول و مقدار سد كننده چپ سلول همسايه سمت راست سلول، برابر يك باشد، مقدار سلول سمت راست جايگزين اين سلول شده و در غير اين صورت جابجايي اتفاق نمي‌افتد. عمليات فوق طبق رابطه (۸) صورت مي‌پذيرد.

$$x_i = \begin{cases} x_{i+1} & S_i^r = 1 \wedge S_{i+1}^l = 1 \\ x_{i-1} & S_i^l = 1 \wedge S_{i-1}^r = 1 \\ x_i & Otherwise \end{cases} \qquad (۸)$$

اگر روابط (۶) و (۷) در رابطه (۸) جايگزين و كمي ساده‌سازي انجام گيرد، رابطه (۹) بدست مي‌آيد. بر اساس رابطه (۹) هر سلول براي جابجايي مقدار خود با مقدار سلول همسايه‌اش نيازمند بررسي محتواي خود با سه سلول همسايه خود (يک سلول سمت چپ و دو سلول سمت راست) مي‌باشد.

$$x_i = \begin{cases} x_{i+1} & if\,(x_i > x_{i+1} \,\&\, x_{i+1} \le x_{i+2}) \\ x_{i-1} & if\,(x_i \le x_{i-1} \,\&\, x_i \le x_{i-1}) \\ x_i & Otherwise \end{cases} \qquad (۹)$$

با اينكه همسايگي تعريف شده در اين الگوريتم شعاع يک متقارن و شامل دو همسايه مي‌باشد ولي به دليل دو مرحله اجرايي q_1 و q_0 برمبناي رابطه (۹)، همسايگي مورد استفاده هر سلول براي مرتب‌سازي، سه سلول همانند شكل (۳) مي‌باشد.

i-1	i	i+1	i+2

شکل(۳): همسايگي واقعي در الگوريتم اول گورديلو_لونا.

اين الگوريتم هنگامي خاتمه مي‌يابد كه چرخه انتقال q_0 به q_1 و q_1 به q_0، $2n-3$ مرتبه تكرار شود. اگر آرايه‌اي در كمتر از $2n-3$ مرحله مرتب شود، نياز به يک كنترل بيروني براي توقف اجراي الگوريتم مي‌باشد. در اين صورت مي‌توان از حافظه‌هاي سد كننده براي كنترل توقف استفاده نمود. به عبارتي اگر هيچ حافظه سد كننده‌اي يک نشده باشد، عمليات مرتب‌سازي مي‌تواند پايان يابد. با راهكار نظارت بيروني، مرتب‌سازي در بدترين شرايط پيچيدگي زماني $2n-3$ خواهد داشت. در جدول (۱)، مرتب‌سازي صعودي اعداد [۱-۱۰] با ترتيب نزولي با الگوريتم اول گورديلو_لونا نشان داده شده است.

جدول(۱): مرتب‌سازي اعداد [۱-۱۰]، الگوريتم گورديلو_لونا.

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	N.
۲	۱	۳	۴	۵	۶	۷	۸	۹	۱۰	۱
۲	۳	۱	۴	۵	۶	۷	۸	۹	۱۰	۲
۳	۲	۴	۱	۵	۶	۷	۸	۹	۱۰	۳
۳	۴	۲	۵	۱	۶	۷	۸	۹	۱۰	۴
۴	۳	۵	۲	۶	۱	۷	۸	۹	۱۰	۵
۴	۵	۳	۶	۲	۷	۱	۸	۹	۱۰	۶
۵	۴	۶	۳	۷	۲	۸	۱	۹	۱۰	۷
5	۶	۴	۷	۳	۸	۲	۹	۱	۱۰	۸
۶	۵	۷	۴	۸	۳	۹	۲	۱۰	۱	۹
۶	۷	۵	۸	۴	۹	۳	۱۰	۲	۱	۱۰
۷	۹	۸	۵	۹	۴	۱۰	۳	۲	۱	۱۱
۷	۸	۶	۹	۵	۱۰	۴	۳	۲	۱	۱۲
۸	۷	۹	۶	۱۰	۵	۴	۳	۲	۱	۱۳
۸	۹	۷	۱۰	۶	۵	۴	۳	۲	۱	۱۴
۹	۸	۱۰	۷	۶	۵	۴	۳	۲	۱	۱۵
۹	۱۰	۸	۷	۶	۵	۴	۳	۲	۱	۱۶
۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۱۷

اولين جابجايي يک آرايه نزولي در الگوريتم اول گورديلو_لونا، جابجايي كوچكترين عدد با همسايه خود مي‌باشد. در الگوريتم دوم گورديلو_لونا با بهبود سد كننده‌هاي چپ و راست، جابجايي بزرگترين عدد با همسايه خود در اولين جابجايي امكان‌پذير شده است. همسايگي تعريف شده در اين الگوريتم با شعاع متقارن يعني دو سلول سمت راست و دو سلول سمت چپ مي‌باشد. با اين وجود با توجه به دو وضعيت q_1 و q_0 تعداد سلول همسايه مورد استفاده هر سلول براي مرتب‌سازي، پنج سلول (سه سلول سمت راست و دو سلول سمت چپ) همانند شكل (۴) مي‌باشد.

i-2	i-1	i	i+1	i+2	i+3

شكل(۴): همسايگي واقعي در الگوريتم دوم گورديلو_لونا.

در اين الگوريتم پيچيدگي زماني در بدترين شرايط متناسب با $\lceil 3n/2 - 2 \rceil$ مي‌باشد.

۳-۲- الگوريتم‌هاي مرتب‌سازي خطي پيشنهادي

در اين قسمت دو الگوريتم مرتب‌سازي پيشنهادي جديد با استفاده از اتوماتاي سلولي خطي معرفي مي‌شوند. در اين الگوريتم‌ها نحوه جابجايي از راست مي‌باشد، به عبارتي جابجايي بر اساس مقايسه هر سلول و سلول‌هاي سمت راست آن صورت مي‌گيرد.

۳-۲-۱- الگوريتم اول مرتب‌سازي با اتوماتاي سلولي خطي

در اين الگوريتم همسايگي برابر دو و به صورت نامتقارن تعريف شده است. همسايه هر سلول i شامل، اولين همسايه سمت راست $i+1$ (همسايه چپ $i-1$) و دومين همسايه سمت راست $i+2$ (دومين همسايه چپ $i-2$) آن مي‌باشد. مي‌توان همسايگي را از چپ و يا از راست تعريف نمود كه در ادامه همسايگي از سمت راست همانند شكل (۵) در نظر گرفته شده است.

i	i+1	i+2

شكل(۵): مرتب‌سازي با همسايگي نامتقارن دو.

بر اساس شكل (۵) هر سلول با دو همسايه سمت راست (چپ) خود در تعامل بوده و جابجايي بين سلول i-ام و سلول i+1-ام، بر اساس محتواي همسايگي اتفاق مي‌افتد. در ابتدا هر سلول محتواي خود را با سلول سمت راست خود مقايسه نموده و در نتيجه محتواي اين سلول بزرگتر يا كوچكتر از سلول همسايه سمت راست خود خواهد بود. به عبارتي در يك همسايگي، يك مقايسه بين محتواي سلول i و$i+1$ و يك مقايسه بين محتواي سلول $i+1$ و $i+2$ اتفاق مي‌افتد. از نتايج اين دو مقايسه انتقال و يا عدم انتقال بين محتواي دو سلول i و$i+1$ مشخص مي‌گردد.

براي بيان قانون حاكم بر همسايگي، از روش نمايش خاصي شامل دو بيت استفاده مي‌نماييم. در اين نحوه نمايش، نتايج مقايسات صورت گرفته در هر همسايگي با دو بيت كه هر بيت بيانگر يك مقايسه مي‌باشد، نشان داده مي‌شود. اين نحوه نمايش مقايسات به صورت بيت‌هاي مقايسه‌اي تنها براي نشان دادن قانون حاكم بر همسايگي سلول‌هاي اتوماتاي سلولي در نظر گرفته شده‌اند. براي نمونه اگر محتواي سلول i بزرگتر از محتواي سلول $i+1$ باشد، مقدار يك در غير اين صورت مقدار صفر براي بيت سمت چپ منظور مي‌گردد. محتواي بيت سمت راست نيز از مقايسه بين محتواي سلول‌هاي $i+1$ و$i+2$ بدست مي‌آيد. مرتب‌سازي در اين روش براي سلول i از قوانين ساده محلي زير طبعيت مي‌نمايد:

- در بيت‌هاي مقايسه‌اي به صورت $\boxed{0\,|\,0}$، $\boxed{0\,|\,1}$ و يا $\boxed{1\,|\,1}$ هيچ جابجايي صورت نمي‌گيرد.
- بيت‌هاي مقايسه‌اي به صورت $\boxed{1\,|\,0}$ باعث جابجايي محتواي سلول i و سلول $i+1$ مي‌شوند.

هرگاه طبق قوانين حاكم بر اتوماتاي سلولي، شرط جابجايي سلول مورد بررسي كه از قانون بيت‌هاي مقايسه‌اي طبعيت مي‌نمايد، ارضاء شود، محتواي سلول با محتواي سلول سمت راست آن تعويض مي‌شود.

۳-۲-۲- الگوريتم پيشنهادي اول

مرتب‌سازي درالگوريتم پيشنهادي اول به صورت زير انجام مي‌شود. در ابتدا در هر همسايگي محتواي دو سلول i و$i+1$ و دو سلول $i+1$ و$i+2$ مقايسه مي‌شوند (محاسبه بيت‌هاي مقايسه‌اي). جابجايي بين سلول i و$i+1$ تنها هنگامي رخ مي‌دهد كه محتواي سلول i بزرگتر از محتواي سلول $i+1$ بوده و محتواي سلول $i+1$ كوچكتر از محتواي سلول $i+2$ باشد؛ در غير اين صورت جابجايي صورت نمي‌پذيرد. جابجايي محتواي دو سلول i و$i+1$ در الگوريتم اول توسط رابطه (۱۰) نشان داده شده‌است.

$$if\ \ (x_i > x_{i+1}) \wedge (x_{i+1} < x_{i+2})\ \{swap(x_i, x_{i+1})\}$$
$$else\quad \{nothing\} \tag{۱۰}$$

در را بطه فوق محتواي سلول i با x_i و سلول $i+1$ با x_{i+1} نمايش داده شده است. تابع swap، براي جابجايي محتواي دو سلول مي‌باشد.

در اين الگوريتم يك آرايه n عنصري در بدترين شرايط (ترتيب نزولي اعداد) در $2n-3$ مرحله به صورت صعودي مرتب مي‌شود. در جدول (۲)، مرتب‌سازي اعداد [۱۰-۱] در بدترين شرايط با الگوريتم پيشنهادي اول نشان داده شده است.

جدول(۲): مرتب‌سازي اعداد با الگوريتم پيشنهادي اول.

۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰	N.
۲	۱	۳	۴	۵	۶	۷	۸	۹	۱۰	۱
۲	۳	۱	۴	۵	۶	۷	۸	۹	۱۰	۲
۳	۲	۴	۱	۵	۶	۷	۸	۹	۱۰	۳
۳	۴	۲	۵	۱	۶	۷	۸	۹	۱۰	۴
۴	۳	۵	۲	۶	۱	۷	۸	۹	۱۰	۵
۴	۵	۳	۶	۲	۷	۱	۸	۹	۱۰	۶
۵	۴	۶	۳	۷	۲	۸	۱	۹	۱۰	۷
5	۶	۴	۷	۳	۸	۲	۹	۱	۱۰	۸
۶	۵	۷	۴	۸	۳	۹	۲	۱۰	۱	۹
۶	۷	۵	۸	۴	۹	۳	۱۰	۲	۱	۱۰
۷	۹	۸	۵	۹	۴	۱۰	۳	۲	۱	۱۱
۷	۸	۶	۹	۵	۱۰	۴	۳	۲	۱	۱۲
۸	۷	۹	۶	۱۰	۵	۴	۳	۲	۱	۱۳
۸	۹	۷	۱۰	۶	۵	۴	۳	۲	۱	۱۴
۹	۸	۱۰	۷	۶	۵	۴	۳	۲	۱	۱۵
۹	۱۰	۸	۷	۶	۵	۴	۳	۲	۱	۱۶
۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۱۷

اگر بخواهيم تعداد مراحل مرتب‌سازي الگوريتم در بدترين و بهترين شرايط يكسان نباشد و بر اساس عناصر موجود در آرايه تغيير نمايد، نيازمند یک ناظر بيروني براي خاتمه دادن به اجراي الگوريتم مي‌باشيم. هرگاه جابجايي بين هيچ سلولي رخ ندهد، آرايه مرتب شده و الگوريتم مي‌تواند خاتمه يابد.

۳-۲-۳- الگوريتم مرتب‌سازي خطي پيشنهادي دوم

الگوريتم پيشنهادي دوم با گسترش شعاع همسايگي الگوريتم پيشنهادي اول ايجاد شده است. اين الگوريتم با دو نوع همسايگي مختلف پياده‌سازي شده است. در پياده‌سازي اول همسايگي با شعاع دو و به صورت متقارن مي‌باشد. به عبارتي براي هر سلول i، همسايگي شامل دو همسايه راست $i+1$ و$i+2$ و دو همسايه چپ $i-1$ و$i-2$ مي‌باشد. در پياده‌سازي دوم همسايگي با شعاع دو و به صورت نامتقارن مي‌باشد. به عبارتي براي هر سلول i، همسايگي شامل سه

همسایه راست $i+1$، $i+2$ و $i+3$ و يك همسايه چپ $i-1$ تعريف شده است. همسايگي‌هاي اين روش در شكل (۶) نشان داده شده‌اند.

i-1	**i**	i+1	i+21	i+3

i-2	i-1	**i**	i+11	i+2

شكل(۶): مرتب‌سازي با شعاع همسايگي دو متقارن و نامتقارن.

در پیاده‌سازي با همسايگي متقارن، هر سلول با دو همسايه سمت راست و دو همسايه سمت چپ خود در تعامل بوده و بر اساس محتواي آنها جابجايي بين سلول i-ام و سلول i+1-ام اتفاق مي‌افتد. همچنين در پیاده‌سازي با همسايگي نامتقارن هر سلول با سه همسايه سمت راست و يك همسايه سمت چپ خود در تعامل بوده و بر اساس محتواي آنها جابجايي بين سلول i-ام و سلول i+1-ام اتفاق مي‌افتد. هر سلول محتواي خود را تنها با سلول سمت راست خود مقايسه نموده و در نتيجه محتواي سلول مذكور يا بزرگتر از سلول همسايه سمت راست خود خواهد بود و يا كوچكتر از آن. بر اساس اين مقايسات امكان جابجايي محتواي دو سلول i-ام و سلول i+1-ام بررسي مي‌شود. مرتب‌سازي در اين روش براي سلول i در همسايگي متقارن از قوانين ساده محلي زير طبعيت مي‌نمايد:

- در بيت‌هاي مقايسه‌اي به صورت [x][x][۰][x]، [x][۱][۱][۱] و [۰][۱][۱][۰] هيچ جابجايي صورت نمي‌گيرد.
- بيت‌هاي مقايسه‌اي به صورت [x][۰][۱][x] و [۱][۱][۱][۰] باعث جابجايي محتواي سلول i و سلول $i+1$ مي‌شوند.

در قوانين فوق x به معناي مهم نبودن محتواي مقايسه مي‌باشد. هرگاه طبق قوانين فوق، شرط جابجايي سلول مورد بررسي كه از قانون بيت‌هاي مقايسه‌اي طبعيت مي‌نمايد، ارضاء شود، محتواي سلول مركزي با محتواي سلول سمت راست آن تعويض مي‌شود.

۳-۲-۴- مراحل اجراي الگوريتم پيشنهادي دوم

مرتب‌سازي درالگوريتم پيشنهادي دوم به صورت زير انجام مي‌شود. در ابتدا در هر همسايگي محتواي هر دو سلول مجاور با يكديگر مقايسه مي‌شود (محاسبه بيت‌هاي مقايسه‌اي). جابجايي بين سلول i و $i+1$ طبق رابطه (۱۱) براي همسايگي متقارن، و رابطه (۱۲) براي همسايگي نامتقارن تغيير مي‌نمايد.

$$\begin{array}{l} if\ \ ((x_i > x_{i+1}) \wedge (x_{i-1} < x_i)) \vee \\ ((x_{i-2} > x_{i-1}) \wedge (x_{i-1} > x_i) \wedge \\ (x_i > x_{i+1}) \wedge (x_{i+1} < x_{i+2})) \\ \{swap(x_i, x_{i+1})\} \\ else \quad \{nothing\} \end{array} \qquad (۱۱)$$

$$\begin{array}{l} if\ \ ((x_i > x_{i+1}) \wedge (x_{i+1} < x_{i+2})) \vee \\ ((x_i > x_{i-1}) \wedge (x_i > x_{i+1}) \wedge \\ (x_{i+1} > x_{i+2}) \wedge (x_{i+2} < x_{i+3})) \\ \{swap(x_i, x_{i+1})\} \\ else \quad \{nothing\} \end{array} \qquad (۱۲)$$

در رابطه فوق محتواي سلول i با x_i و سلول $i+1$ با x_{i+1} نمايش داده شده است. تابع swap، براي جابجايي محتواي دو سلول i و $i+1$ مي‌باشد.

اگر بخواهيم تعداد مراحل مرتب‌سازي الگوريتم در بدترين و بهترين شرايط يكسان نباشد و بر اساس عناصر موجود در آرايه تغيير نمايد، همانند الگوريتم پيشنهادي اول نيازمند افزودن نظارت بيروني مي‌باشيم. اين نشارت بيروني مي‌تواند شامل یک حافظه مشترک براي تمام سلول‌ها باشد؛ به اين نحو كه هر سلول اگر محتوايش با سلول كنارش جابجا شود، در اين حافظه مقدار یک قرار دهد.

در اين الگوريتم يك آرايه n عنصري در بدترين شرايط (ترتيب نزولي اعداد) در $\lceil 3n/2-2 \rceil$ مرحله به ترتيب صعودي مرتب مي‌شود. تعداد مراحل مرتب‌سازي الگوريتم پيشنهادي دوم با هر دو پیاده‌سازي (همسايگي متقارن و نامتقارن) در بدترين شرايط يكسان مي‌باشد، ولي در حالت كلي تعداد مراحل مرتب‌سازي اعداد آرايه‌هايي كه در بدترين شرايط نيستند، با اين دو پیاده‌سازي يكسان نمي‌باشد. به عبارتي تعداد مراحل اجرايي اين دو پیاده‌سازي در آرايه‌هاي مختلف مي‌تواند متفاوت باشد. در جدول (۳)، مرتب‌سازي اعداد [۱۰-۱] در بدترين شرايط با الگوريتم پيشنهادي دوم نشان داده شده است.

جدول(۳): مرتب‌سازي اعداد با الگوريتم پيشنهادي دوم، اتوماتاي سلولي با شعاع همسايگي دو متقارن/ نامتقارن (چهار همسايه)

N.	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱
۱	۹	۱۰	۸	۷	۶	۵	۴	۳	۱	۲
۲	۹	۸	۱۰	۷	۶	۵	۴	۱	۳	۲
۳	۸	۹	۷	۱۰	۶	۵	۱	۴	۲	۳
۴	۸	۷	۹	۶	۱۰	۱	۵	۲	۴	۳
۵	۷	۸	۶	۹	۱	۱۰	۲	۵	۳	۴
۶	۷	۶	۸	۱	۹	۲	۱۰	۳	۵	۴
۷	۶	۷	۱	۸	۲	۹	۳	۱۰	۴	۵
۸	۶	۱	۷	۲	۸	۳	۹	۴	۱۰	۵
۹	۱	۶	۲	۷	۳	۸	۴	۹	۵	۱۰
۱۰	۱	۲	۶	۳	۷	۴	۸	۵	۹	۱۰
۱۱	۱	۲	۳	۶	۴	۷	۵	۸	۹	۱۰

۱۲	۱	۲	۳	۴	۶	۵	۷	۸	۹	۱۰
۱۳	۱	۲	۳	۴	۵	۶	۷	۸	۹	۱۰

۴- نتيجه گيري

در اين مقاله دو الگوريتم مرتب‌سازي براي اتوماتاي سلولي خطي پيشنهاد گرديد. تعدا مراحل مورد نياز براي مرتب‌سازي در الگوريتمهاي پيشنهادي حدودا به نصف تعداد مراحل مورد نياز الگوريتم گورديلو_اونا كه تنها الگوريتم مرتب سازي گزارش شده براي اتوماتاي سلولي خطي كمتر ميباشد ميرسد. شعاع همسايگي هر دو الگوريتم پيشنهادي در مقايسه با شعاع همسايگي الگوريتم گورديلو-لونا كمتر و بهمين دليل محاسبات مورد نياز براي جابجايي مقدار دو سلول كمتر مي‌باشد. همچنين قوانين استفاده شده در الگوريتم‌هاي پيشنهادي بمراتب ساده‌تر از قوانين معرفي شده توسط الگوريتم‌هاي گورديلو-لونيا مي‌باشد.

مراجع

[1] Wolfram, S., "Computation Theory of Cellular Automata", Physica Scripta, Vol. 9, pp. 170-183, 1985.

[2] Megson, G.M., *An Introduction to Systolic Algorithm Design*، Clara don press Oxford، 1992.

[3] Knuth, D.E., *The Art of Computer Programming: Sorting and Searching*, Addison Wesley, 1973.

[4] Akl, S. G., *Parallel Sorting Algorithms*, Orlando, FL: Academic، 1985.

[5] Toffoli, T., Margolus, N., *Cellular Automata Machines: A New Environment for Modeling*, MIT Press Series in Scientific Computation, 1987.

[6] Chen, C.Y.R., Hou, C. Y. and Singh, U., *"Optimal Algorithms for Bubble Sort Based Non-Manhattan Channel Routing"*, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 13, No. 5, pp. 603 – 609, May 1994.

[7] Thompson, D., Kung, H.T., *"Sorting on a Mesh Connected Parallel Computer",* Communication of ACM, Vol. 20, pp. 263-271, 1977.

[8] Batcher, K.E., *"Sorting Network and Their Applications*", AFIP Proc, Vol. 32, pp. 307-314, 1968.

[9] Kummar, M. and Hirschberg, D.S., "*An Efficient Implementation of Batcher's Odd-Even Merge Algorithms and its Application to Parallel Sorting Schemes",* IEEE Transaction on Comput C-32, pp. 254-264, 1983.

[10] Gordillo, L. and Luna, V., *"Parallel Sort on a Linear Array of Cellular Automata",* IEEE Trans. Computer vol 2, pp. 1904-1910, 1994.

[11] Sarkar, P., "*Brief History of Cellular Automata*", ACM Computing Survey, Vol. 32, No 1, 2000.

زيرنويس‌ها

1 Odd-Even Transposition Sort
2 Quick Sort
3 Selection Sort
4 Merge Sort
5 Cellular Automata
6 Systolic Array
7 General
8 Totalistic
9 Outer Totalistic