

طراحی و پیاده‌سازی رفع کننده برخورد در یک سیستم تصمیم‌همیار هوشمند

مبتنی بر نوع نقطه نظرات

فائزه بنکداری^{۱*}، شهرام جعفری^۲

اطلاعات مقاله	چکیده
واژگان کلیدی: رفع برخورد، سیستم تصمیم‌همیار هوشمند، بازی راهکارک بلادرنگ.	رفع برخورد، یک رویه مهم در بسیاری از سیستم‌های هوشمند از جمله سیستم‌های مبتنی بر قانون می‌باشد. این رویه، ترتیب اجرای قوانین را در شرایطی که بیش از یک قانون برای اجرا وجود دارد، تعیین می‌کند. هدف از این مقاله، ارائه یک سیستم تصمیم‌همیار هوشمند است که در هنگام وقوع برخورد، با در نظر گرفتن خطوط استنتاج مجزا برای هر یک از قوانین موجود در مجموعه برخورد، امکان آگاهی از تمامی انتخاب‌های ممکن را برای تصمیم‌گیرنده فراهم می‌سازد. این ایده، در سیستمی که به منظور یاری رساندن به یک بازیکن در یک بازی راهکارک بلادرنگ پیاده‌سازی شده، اعمال شده است. نتایج به‌دست آمده حاکی از آن است که بازیکن به کمک این دستیار هوشمند می‌تواند عملکرد بهتری نسبت به سایر بازیکنان داشته باشد.

۱- مقدمه

به طور کلی، اخذ تصمیم یکی از مهم‌ترین و حساس‌ترین فعالیت‌هایی است که در هر سازمان‌دهی و یا تشکیلاتی صورت می‌گیرد [۲]. برای پشتیبانی و حمایت از این روند پیچیده، دسته متنوعی از سیستم‌های اطلاعاتی مستقل به نام سیستم‌های تصمیم‌همیار، در طی دو دهه گذشته به وجود آمده‌اند. این سیستم‌ها به صورت ابزارهای مبتنی بر کامپیوتری که به منظور پشتیبانی از روند پیچیده اخذ تصمیم و حل مسئله ایجاد می‌شوند، تعریف و در جهت ایجاد محیطی برای تحلیل مسائل، ساخت مدل‌ها و شبیه‌سازی رویه تصمیم‌گیری و برنامه‌های تصمیم‌گیرندگان طراحی می‌گردند [۳]. اگرچه سیستم‌های تصمیم‌همیار با به‌کار بردن منابع اطلاعاتی و ابزارهای تحلیل گوناگون، شرایط بهتر و با کیفیت بیشتری را برای تصمیم‌گیرندگان فراهم می‌سازند و داشتن یک نقش حمایتی به جای جایگزینی کامل افراد در روند اخذ

مسائل بسیاری وجود دارند که محدود به یک راه‌حل منحصر به فرد نمی‌باشند. علاوه بر این، برخی از مسائل ممکن است تعداد نامحدودی مسیر پاسخ متمایز داشته باشند. یک برخورد هنگامی روی می‌دهد که قوانینی که با حقیقت‌های یکسانی فعال شده‌اند، نتایج متفاوتی را تولید نمایند. سیستم‌های هوشمند از قبیل سیستم‌های مبتنی بر قانون، ابزارهای برنامه‌ریزی و ساختارهای وابسته به دانش، از راهکارهای متفاوتی برای رفع برخورد استفاده می‌کنند [۱]. در این مقاله، به منظور رفع برخورد در یک سیستم مبتنی بر قانون، از پیگیری نقطه نظرات مختلف در مسیرهای استنتاج مجزا استفاده شده است.

* پست الکترونیک نویسنده مسئول: faeze.bonakdari@gmail.com

۱. کارشناس ارشد مهندسی کامپیوتر از دانشگاه شیراز

۲. استادیار، دانشکده مهندسی برق و کامپیوتر، دانشگاه شیراز

جمله مهمترین وظایفی که توسط موتور استنتاج صورت می‌گیرد، رفع برخورد است [۷]. به طور کلی، رفع برخورد، یک راهکار برای انتخاب ترتیب اجرای قوانین است، هنگامی که بیش از یک قانون بتواند اجرا شود.

برای رفع برخورد، روش‌های مختلفی وجود دارد. ساده‌ترین راه‌حل، انتخاب تصادفی و یا ترتیبی قوانین است. در برخی از راهکارها، از جمله مهمترین فاکتورهایی که در انتخاب قوانین مؤثر است، مقدار اولویتی است که توسط سازنده سیستم به هر قانون اختصاص داده می‌شود که در این روش برای قوانین با الویت یکسان باید از روش دیگری مانند اجرای قانونی که به تازگی فعال شده، و یا اجرای قانونی که پیش از سایر قوانین فعال شده است، استفاده گردد [۷].

راهکارهایی نظیر LEX^۲ و MEA^۳، ترتیب اجرای قوانین با اولویت یکسان را با توجه به تأخر فعال‌سازی الگوهای موجود در قوانین، تعیین می‌کنند [۸].

روش‌های خیره‌تر (که دارای سطح بالاتری هستند) از اطلاعات آماری مرتبط با موفقیت‌ها و عدم موفقیت‌های پیشین در هنگام به‌کار بردن قوانین مختلف، به منظور پی بردن به احتمال موفقیت، استفاده می‌کنند. همچنین برخی از روش‌ها، هزینه‌های قوانین را که نشان‌دهنده تلاش‌هایی است که حل‌کننده مسئله برای انجام اعمال بدان نیازمند است (مانند زمان) به حساب می‌آورند [۱].

روش‌های آماری (مبتنی بر تئوری بیز) اثبات کرده‌اند که نه تنها برای رفع برخورد، بلکه برای مدل‌سازی برخی از جنبه‌های رفتار انسان نیز مفید هستند. به عنوان مثال، ساختار وابسته به دانش ACT-R، از اطلاعات آماری زیرنمادی برای انتخاب یک قانون از میان مجموعه بسیاری از قوانینی که با هدف جاری مطابقت دارند (مجموعه برخورد)، استفاده می‌کند. در این روش، به هر قانون یک مقدار سودمندی نسبت داده می‌شود و در هنگام وقوع یک برخورد، قانون با بیشترین مقدار

تصمیم، از اهداف اصلی آنهاست [۴]، اما نمی‌توان آنها را به عنوان یک دستیار هوشمند برای تصمیم‌گیرندگان در نظر گرفت. سیستم‌های تصمیم‌همیار هوشمند برای مسائل عمومی که به تصمیم‌گیری‌های مکرر نیاز دارند مفید و از لحاظ اقتصادی مقرون به صرفه هستند. این سیستم‌های محاوره‌ای مبتنی بر کامپیوتر، برای حل مسائل نیمه ساخت یافته، از ترکیب داده و دانش تخصصی و مدل‌هایی که برای حمایت از تصمیم‌گیرندگان در سازمان‌ها به کار می‌رود، با تکنیک‌های هوش مصنوعی استفاده می‌کنند [۵]. در این مقاله، طراحی و پیاده‌سازی یک سیستم تصمیم‌همیار هوشمند برای یک بازی راهکارک بلادرنگ^۱، شرح داده شده است.

تحقیقات در زمینه بازی در حیطه هوش مصنوعی، در سال‌های اخیر تمرکز خود را از بازی‌های تخته‌ای مانند شطرنج و یا Go، به بازی‌های راهکارک بلادرنگ به عنوان زمینه‌ای به منظور بررسی و آموزش رفتارهای پیچیده، تغییر داده است. توجه هوش مصنوعی به این نوع از بازی‌ها، به دلیل سطوح پیچیدگی موجود در آنها مانند مدیریت منابع و اخذ تصمیم در شرایط غیرقطعی می‌باشد [۶].

این مقاله شامل ۶ بخش است، که در بخش ۲ آن کارهای انجام شده تاکنون بیان می‌شود. بخش ۳ ساختار سیستم طراحی شده را شرح می‌دهد. در بخش ۴ نحوه پیاده‌سازی و نتایج به‌دست آمده ارائه شده و بخش ۵ نیز به نتیجه‌گیری و کارهای آینده می‌پردازد. در انتها نیز مراجع مورد استفاده در این کار عنوان شده‌اند.

۲- کارهای پیشین

در بسیاری از سیستم‌های مبتنی بر قانون، موتور استنتاج، یک مؤلفه نرم‌افزاری است که در هنگام اجرای برنامه کاربردی، روی مجموعه‌ای از قوانین استنتاج می‌کند. از

² Lexicographic Sort

³ Means-Ends-Analysis

¹ Real-Time Strategy Game

بازی تیراندازی با عنوان Quake II [۱۹] استفاده کرده‌اند.

۳- ساختار سیستم پیشنهادی

در سیستم‌هایی که رفع برخورد در روند استنتاج آنها صورت می‌گیرد، موتور استنتاج یک رویه سه مرحله‌ای شناسایی- رفع - عمل را روی مجموعه قوانین اعمال می‌کند [۲۰]:

- شناسایی: تشخیص قوانینی که می‌توانند اجرا گردند و قرار دادن آنها در مجموعه برخورد.
- رفع: استفاده از یک راهکار به منظور انتخاب یک قانون از مجموعه برخورد.
- عمل: اجرای قانون منتخب و افزودن نتایج آن به حافظه کاری^۱.

ساختار سیستمی که در این مقاله طراحی و پیاده‌سازی آن شرح داده شده، در شکل ۱ نشان داده شده است. این سیستم، یک سیستم تصمیم‌همیار هوشمند است که در مرحله‌ی رفع برخورد در طی روند استنتاج خود، با در نظر گرفتن یک سیر (خط) استنتاج جداگانه برای هر یک از قوانین موجود در مجموعه برخورد، تمامی حالات ممکن برای اولویت‌بندی در اجرای قوانین را تحت پوشش قرار داده و این امکان را برای تصمیم‌گیرنده فراهم می‌سازد که به همه انتخاب‌های ممکن خود، آگاه گردد، و تصمیم‌گیرنده را در اخذ بهترین تصمیم با توجه به شرایط موجود یاری می‌رساند. این سیستم در واقع نقش یک دستیار هوشمند را برای تصمیم‌گیرنده ایفا می‌نماید و هدف از طراحی آن، برخلاف یک سیستم خبره، جایگزینی فرد تصمیم‌گیرنده نمی‌باشد.

سودمندی انتخاب می‌گردد [۹]. در بهبودی که بر این روش مطرح شده [۱]، از اطلاعات آماری مشابه، اما به طریق دیگری استفاده شده است. این شیوه با یک محیط متغیر، وفق‌پذیرتر بوده و پویایی‌های آن نتیجه کاهش بی-نظمی در مدت حل مسئله می‌باشد. این روش بسیاری از پارامترهای موجود در ACT-R را، با الهام از اطلاعات نظری مربوط به مدل‌های شناختی اخذ تصمیم، یادگیری و احساسات، اصلاح و ترکیب نموده و با برآورد نرخ یک روند پواسن فرضی، هزینه یک راهکار بخصوص را تخمین می‌زند. برخورد، با انتخاب راه‌حلی با کمترین هزینه پیش-بینی شده، رفع می‌گردد.

تحقیق در زمینه راهکار و اخذ تصمیم کاراکترهای بازی در هوش مصنوعی، از طراحی حریفان هوشمند در بازی-هایی با دو بازیکن مانند checkers و Othello پدیدار شده است [۱۰-۱۲]. به ویژه در زمینه Othello ثابت شد که حریفانی که توسط کامپیوتر کنترل می‌گردند نه تنها برای رقابت، بلکه می‌توانند برای شکست بازیکنان واقعی طراحی گردند [۱۳].

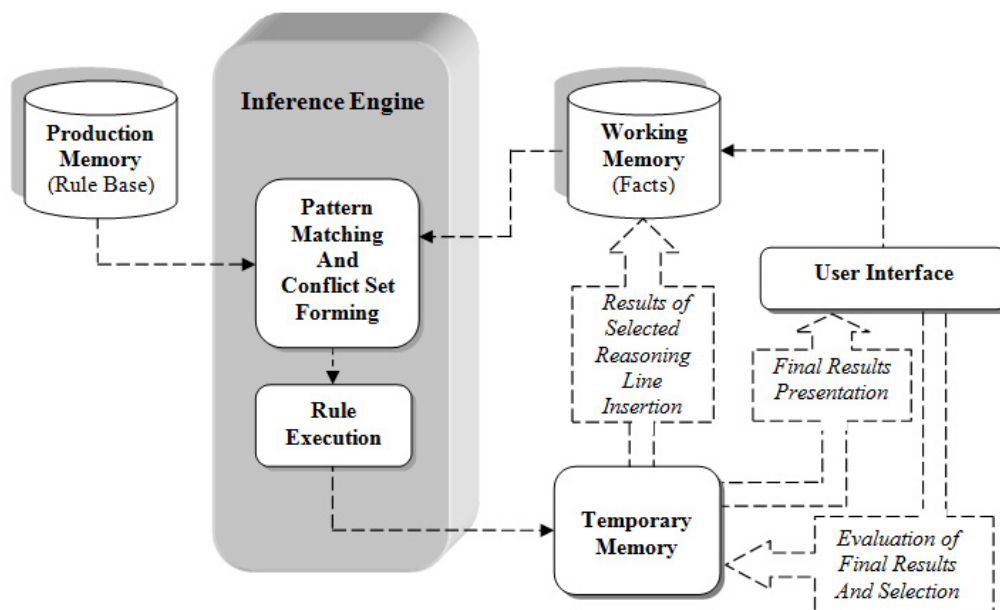
علاوه بر این، سیستم‌هایی مبتنی بر قانون در جهت یاری رساندن به بازیکن در بازی‌هایی نظیر Minesweeper و Sudoku نیز طراحی شده‌اند [۱۴ و ۱۵].

Buro و همکاران بر این باورند که بازی‌های استراتژیک بلادرنگ، محیط خوبی را برای بررسی ایده‌های بسیاری در زمینه‌هایی مانند برنامه‌ریزی بلادرنگ، اخذ تصمیم در شرایط غیر قطعی، مدل‌سازی حریف، مدیریت منابع و مسیریابی فراهم می‌سازند. تحقیقات هوش مصنوعی در زمینه این نوع از بازی‌ها، از حمایت‌های وسیع برخی از مؤسسات دفاعی و نیز تولیدکنندگان تجاری بازی‌ها برخوردار هستند [۱۶].

با در نظر گرفتن محیط‌های پیچیده بازی‌های تجاری، Ponsen و Spronk [۱۷]، یک شبکه را برای بازی‌های و اتصال حالات مجازی، برای بازی Wargus ارائه داده‌اند (Wargus یک بازی استراتژیک بلادرنگ نسبتاً پیچیده و هجوی بر بازی تجاری مشهور WarCraft II می‌باشد).

سایر محققان از این شبکه برای مدیریت یک شهر بلادرنگ و جلوگیری از اشتباهات مکرر در بازی SimCity [۱۸]، و ایجاد یک حریف کامپیوتری در یک

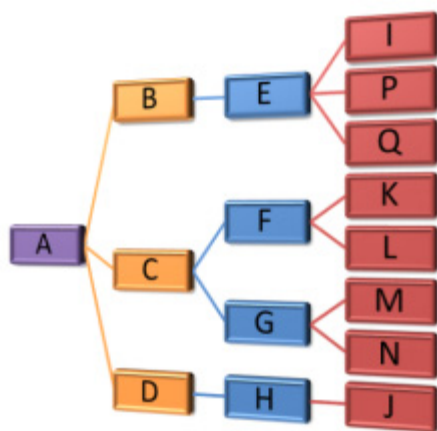
¹ Working Memory



شکل ۱- ساختار سیستم پیشنهادی

در نظر می‌گیرد، نشان داده شده است. مسیرهای استنتاج موجود در این درخت تا انتها پیش برده می‌شود و نتایج حاصل، در یک حافظه موقت ذخیره می‌گردد.

برای شرح ایده رفع برخوردی که در این سیستم به کار برده شده، فرض کنید در این سیستم، مجموعه قوانینی به صورت زیر وجود داشته باشد:



شکل ۲- درخت استنتاجی که از بسط تمامی راه‌حل‌های ممکن به دست آمده است

از آنجا که این سیستم یک سیستم تصمیم‌همیار است و نه تصمیم‌گیرنده، نتایج نهایی به دست آمده از هر خط استنتاج، به فرد تصمیم‌گیرنده اطلاع داده شده و از وی خواسته می‌شود که با توجه به این نتایج، تصمیم خود را اخذ کند. مزیت این ایده رفع برخورد در این است که انتخاب تصمیم‌گیرنده از میان تمامی راه‌حل‌های ممکن صورت خواهد گرفت.

- Rule 1: $A \rightarrow B$
- Rule 2: $A \rightarrow C$
- Rule 3: $A \rightarrow D$
- Rule 4: $B \rightarrow E$
- Rule 5: $C \rightarrow F$
- Rule 6: $C \rightarrow G$
- Rule 7: $D \rightarrow H$
- Rule 8: $E \rightarrow I$
- Rule 9: $E \rightarrow P$
- Rule 10: $E \rightarrow Q$
- Rule 11: $H \rightarrow J$
- Rule 12: $F \rightarrow K$
- Rule 13: $F \rightarrow L$
- Rule 14: $G \rightarrow M$
- Rule 15: $G \rightarrow N$

بدین مفهوم که برقرار بودن A منجر به برقراری B و C و D می‌گردد که این سه با یکدیگر برخورد دارند و انتخاب هر ترکیبی از اولویت‌ها در اجرای این قوانین، نتایج متفاوتی را به دنبال خواهد داشت. در این سیستم، به ازای هر قانون از مجموعه قوانین برخورد، یک خط استنتاج مجزا در نظر گرفته شده و در هر مسیر، فرایند استنتاج تا آنجا که ممکن است، پیش برده می‌شود. در شکل ۲ درخت استنتاجی که سیستم برای این مجموعه از قوانین

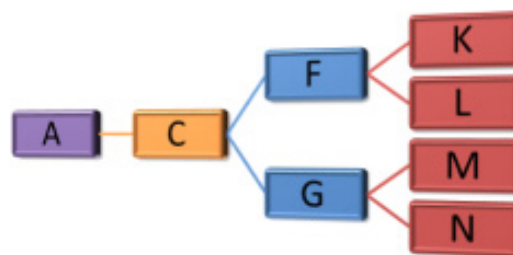
بازی‌ها، بازیکنان علاوه بر کنترل واحدهای خود، با سایر بازیکنان به جنگ می‌پردازند. از اینرو آشنایی با تصمیمات راهبردی گوناگون در هر یک از زمینه‌های فوق، امری ضروری برای بردن بازی به حساب می‌آید.

بازی استراتژیک بلادرنگ مورد استفاده در این بررسی، بازی Age of Empires III (AOE III)، است. این بازی توسط Microsoft Corporation's Ensemble Studios تولید شده و Microsoft Game Studios نسخه‌ی قابل اجرا روی سیستم عامل Windows آن را در ماه‌های پایانی سال 2005 منتشر کرده است. Age of Empires III در سال 2005 در مکان هشتم از لیست پرفروش‌ترین بازی‌های کامپیوتری قرار گرفت و بیش از دو میلیون کپی از آن تا ماه May سال 2008 فروخته شده بود و در سال 2007 با فروش بیش از ۳۱۳ هزار کپی در آن سال، مکان هفتم از لیست پرفروش‌ترین بازی‌های کامپیوتری را به خود اختصاص داد [۲۱].

در طول این بازی، بازیکن در هر لحظه انتخاب‌های بسیاری برای انجام فعالیت‌های ممکن دارد. به عنوان مثال، شناسایی سرزمین‌های ناشناخته، شناسایی مکان قرارگیری دشمن، بررسی منابع مورد نیاز برای ساخت بناهایی که هر یک متعلق به دوره‌ای خاص می‌باشند و جمع‌آوری آن منابع یا تعویض هر یک از منابع با منبع مورد نیاز در آن لحظه، بهبود بناهای موجود، تولید افراد و سلاح‌های جنگی، دفاع از قلمرو در هنگام حمله‌ی دشمن، کنترل راه‌ها و ...

یکی از روش‌هایی که برای انتخاب بهترین کار ممکن در هر لحظه وجود دارد، این است که نتیجه انجام تمامی فعالیت‌های ممکن مشاهده و بر اساس آن تصمیم‌گیری صورت گیرد. موتور استنتاج سیستم تصمیم‌همیار هوشمند مورد بررسی در این مقاله، استنتاج را در هر یک از مسیرهای ممکن تا آنجا که مقادیر منابع بازیکن اجازه می‌دهد، ادامه می‌دهد. به عبارت دیگر در این سیستم، عاملی که منجر به توقف پیشروی درخت استنتاج می‌گردد، مقادیر منابع بازیکن است. در این نرم‌افزار، این

در ادامه‌ی روند استنتاج، اجرای تصمیم مأخوذ می‌تواند منجر به تغییر در ارزش حقایق موجود در حافظه کاری سیستم شود. از این رو شیوه استنتاج در این سیستم مبتنی بر استنتاج غیر یکنواخت می‌باشد. یعنی در طول استنتاج ممکن است ارزش حقیقت‌های موجود تغییر یابد. به عنوان مثال در مجموعه قوانین فرضی فوق، ممکن است پس از آنکه تصمیم‌گیرنده از نتیجه نهایی اجرای هر یک از قوانین مجموعه برخورد آگاه شد، راه‌حل مسیر دوم را به عنوان تصمیم خود انتخاب نماید، که در این صورت، درخت استنتاج سیستم تا این مرحله، همانند شکل ۳ خواهد بود، و عبارات موجود در آن وارد حافظه کاری می‌گردند و در صورت تغییر در شرایط محیط، استنتاج از این نقطه ادامه می‌یابد.



شکل ۳- درخت استنتاج سیستم، پس از اخذ تصمیم توسط تصمیم‌گیرنده

به منظور تحلیل ساختار فوق در یک برنامه کاربردی، یک سیستم تصمیم‌همیار هوشمند برای یک بازی استراتژیک بلادرنگ پیاده‌سازی شده است.

امروزه بازی‌های کامپیوتری تجاری یک جزء در حال رشد از صنعت سرگرمی است و شبیه‌سازی‌ها، یک نمود مهم از آموزش‌های نظامی مدرن به حساب می‌آیند [۱۶].

به‌طور کلی، ایجاد برنامه‌های کامپیوتری در رابطه با بازی-های استراتژیک بلادرنگ، یک زمینه جدید در حیطه بازی در هوش مصنوعی به حساب می‌آید [۱۳]. این بازی‌ها شامل جمع‌آوری و کنترل منابع گوناگون به منظور ساخت و بهبود سریع تعداد زیاد و متنوعی از بناهای پایه و نیروهای مختلف، پیشرفت در فناوری، ایجاد و کنترل واحدهای مختلف نظامی و ... می‌باشند. در این نوع

در این نرم‌افزار (که به صورت همزمان با اجرای بازی، اجرا می‌گردد) پس از ایجاد بازیکن، تمامی انتخاب‌های ممکن برای پیشبرد بازی با توجه به مقادیر منابع، در اختیار وی قرار داده می‌شود. به عنوان مثال، در شروع بازی، تمامی حالات ممکن برای ساخت بناهایی که در این مرحله از بازی و با توجه به منابع موجود امکان‌پذیر است و نیز تمامی حالات ممکن برای بهبود این بناها و آموزش نیروهای انسانی مربوط به هر یک از آنها، به بازیکن ارائه می‌شود. با انتخاب هر یک از این راه‌حل‌ها توسط بازیکن و به‌روز رسانی منابع (که هر دقیقه یکبار انجام می‌پذیرد)، حال تمام راه‌حل‌های ممکن برای ادامه بازی از این مرحله، به بازیکن نشان داده می‌شود.

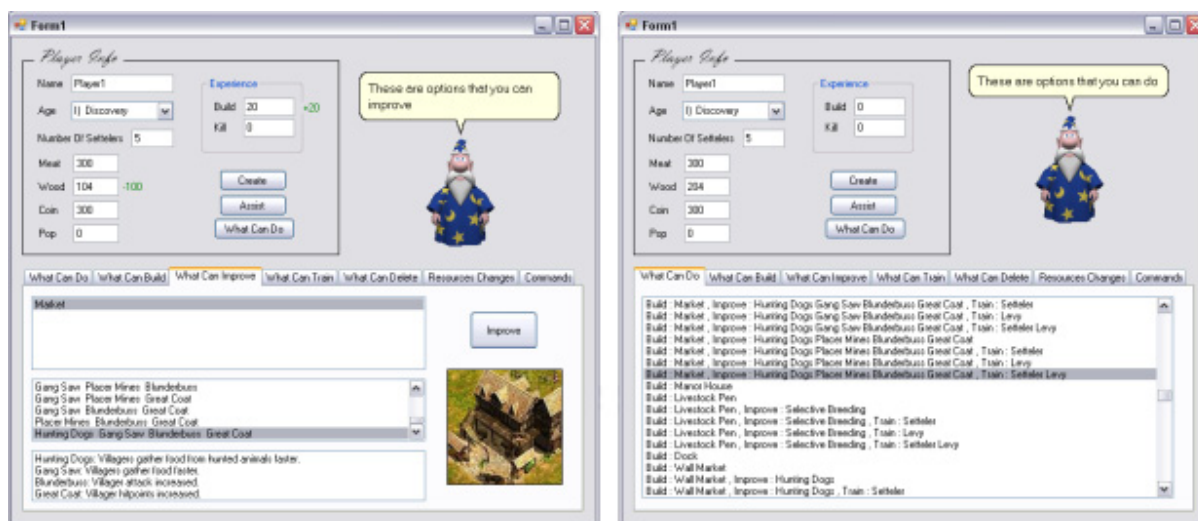
همچنین این نرم‌افزار، با دریافت اطلاعاتی در رابطه با وضعیت کلی بازیکن، وی را در مواردی از قبیل چگونگی توسعه مرزهای قلمرو، چگونگی جمع‌آوری منابع جهت رفتن به مراحل بعدی بازی، چگونگی مبارزه با سایر بازیکنان و... راهنمایی می‌کند. شکل ۴، تصاویری از اجرای نرم‌افزار را نشان می‌دهد.

مقادیر، هر دقیقه یکبار به‌روز رسانی و موجب تغییر در شرایط قبلی بازی می‌گردد و پس از آن، استنتاج مجدداً با بسط تمامی راه‌حل‌های ممکن با توجه به شرایط جدید، پیگیری می‌شود. این موارد، در واقع دلایلی مناسب برای انتخاب این برنامه کاربردی، به منظور بررسی ساختار سیستم پیشنهادی می‌باشند.

۴- پیاده‌سازی و نتایج

سیستم تصمیم‌همیار هوشمندی که ساختار آن در قسمت قبل شرح داده شد، با استفاده از زبان برنامه‌نویسی C# در محیط Visual Studio.NET 2008 پیاده‌سازی شده است.

در این سیستم، ساختار کلی یک قانون، به صورت یک گونه مجازی معرفی و برای هر دسته مرتبط از مجموعه قوانین، یک کلاس که از این کلاس مجازی ارث می‌برد، تعریف شده است.



شکل ۴- تصاویری از اجرای نرم‌افزار

بازیکن با استفاده از راهنمایی‌های این نرم‌افزار به عنوان یک دستیار هوشمند در حین بازی، به دست آورده، با نتایج سایر بازیکنان مقایسه، و در جدول و نمودارهایی که در ادامه می‌آید، نشان داده شده است. معیار این مقایسه

از آنجا که این سیستم نمونه‌ای از یک گونه (منحصر به فرد) است و بر اساس تحقیقات صورت گرفته در این کار، تاکنون سیستمی مشابه با آن طراحی نشده، امکان مقایسه آن با سیستم دیگری امکان‌پذیر نیست. لذا به منظور ارزیابی این سیستم، در ابتدا، نتایجی که یک

همچنین در ادامه، نتایجی که یک بازیکن با استفاده از این نرم‌افزار به دست آورده، با نتایج بازیکنی که در حین اجرای بازی از سیستم دیگری کمک گرفته، نیز مقایسه شده است.

سیستم دوم نیز یک سیستم تصمیم‌همیار هوشمند است که در هنگام وقوع برخورد، از یک توزیع یکنواخت برای انتخاب قوانین جهت اجرا استفاده می‌نماید. در این راهکار احتمال انتخاب هر قانون از مجموعه قوانین برخورد، با یکدیگر برابر می‌باشد. هدف این سیستم تصمیم‌همیار هوشمند، همانند سیستم اول یاری‌رساندن به بازیکن در بازی Age of Empires III می‌باشد.

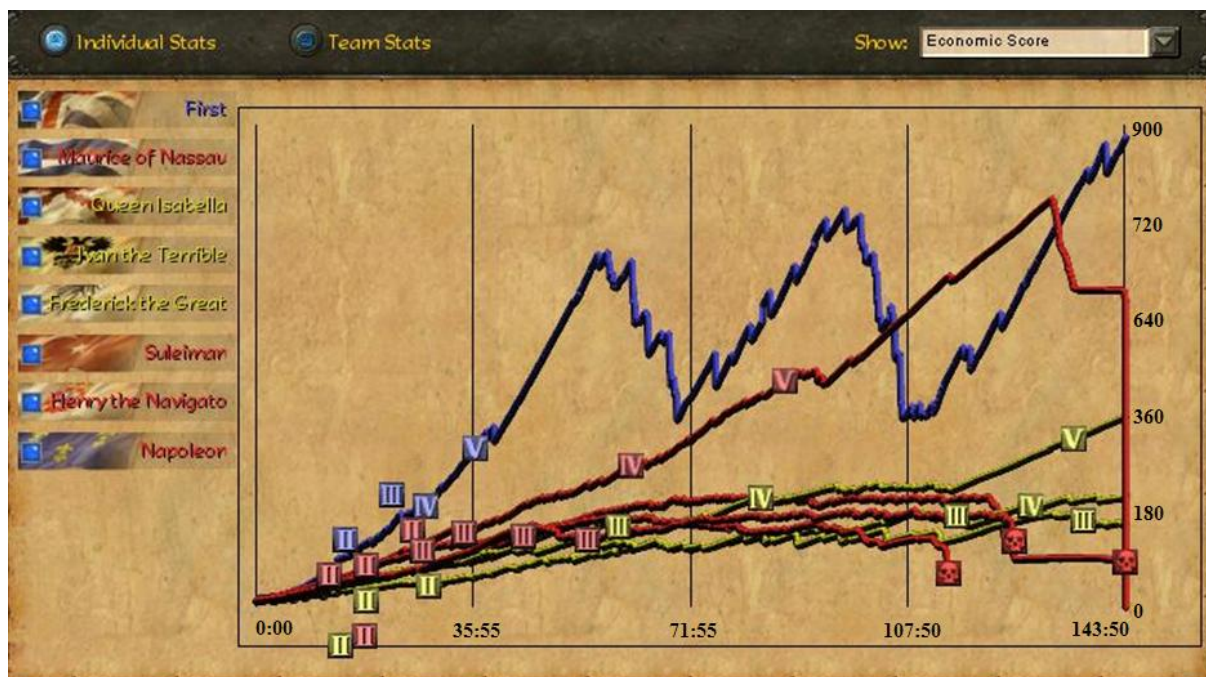
میزان امتیازی است که در پایان بازی برای هر بازیکن محاسبه می‌گردد.

در جدول ۱ و شکل‌های ۵ تا ۸، امتیاز اقتصادی، امتیاز نظامی، تجربه و امتیاز نهایی هشت بازیکن در یک بازی، در مقایسه با یکدیگر نشان داده شده است. (در این جدول و نمودارها، نام بازیکنی است که از این نرم‌افزار در حین بازی استفاده نموده است).

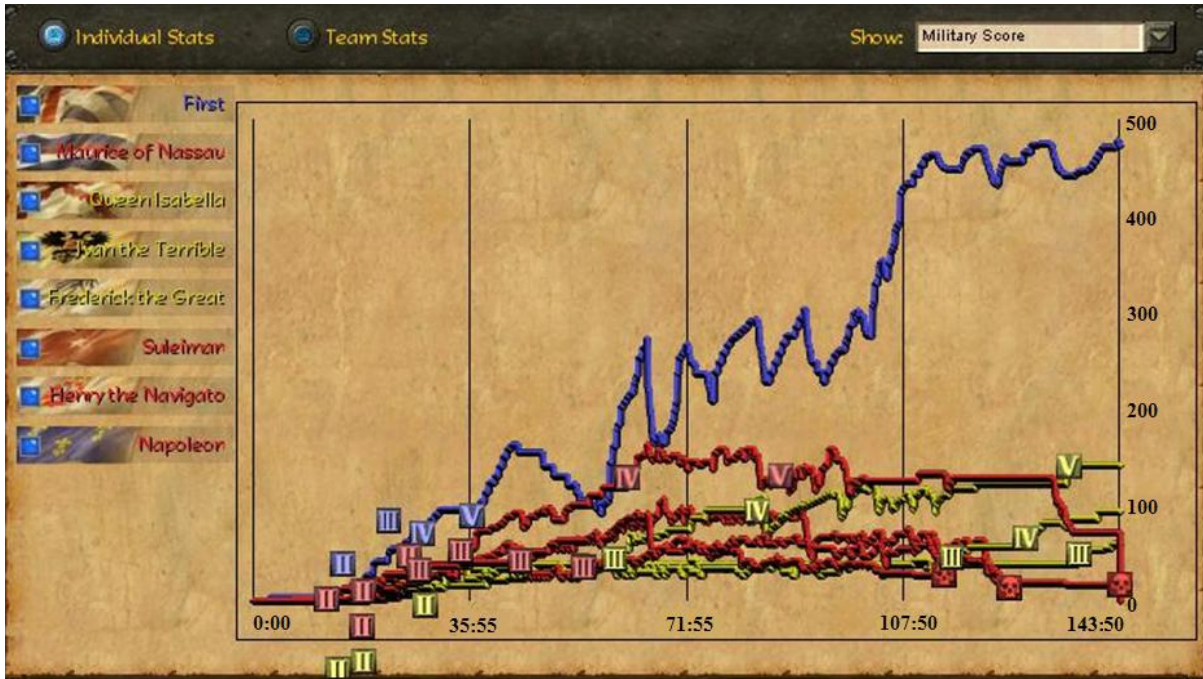
عوامل بسیاری از جمله میزان منابع موجود در انبارها، انواع مختلف بناها و میزان بهبود آن‌ها و فناوری‌هایی که هر بازیکن بدان دست یافته در تعیین این امتیازات مؤثرند، که در همه موارد برتری با بازیکنی است که از این نرم‌افزار استفاده نموده است.

جدول ۱- مقایسه امتیازات هشت بازیکن در یک بازی

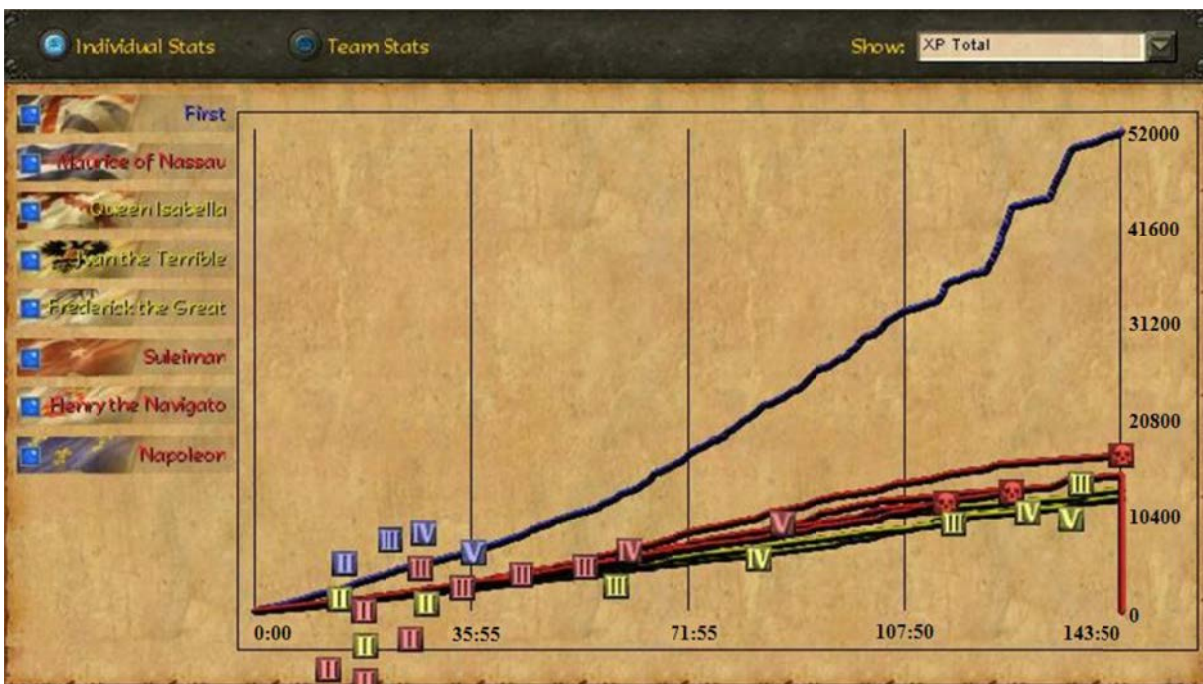
Players Names	Total Score	Experience Awards		
First	1351	Most Resources	280393	First
Maurice of Nassau	104	Least Resources	42390	Henry the Navigator
Queen Isabella	497	Most Military Units	376	First
Ivan the Terrible	307	Most Units Killed	644	First
Frederick the Great	217	Most Units Lost	301	First
Suleiman	642	Highest Score	1351	First
Henry the Navigator	91	Most Treasures	2	Maurice of Nassau
Napoleon	138			



شکل ۵- نمودار امتیاز اقتصادی هشت بازیکن در یک بازی



شکل ۶- نمودار امتیاز نظامی هشت بازیکن در یک بازی



شکل ۷- نمودار تجربه هشت بازیکن در یک بازی



شکل ۸- نمودار امتیاز نهایی هشت بازیکن در یک بازی

برخورد، از ایده تنوع نقطه نظرات استفاده می‌کند، کمک گرفته و Player2 نام یک رقیب کامپیوتری است. در جدول ۳ و در شکل ۹ (ب)، Player1 نام بازیکنی است که از سیستم تصمیم‌همیار هوشمندی که جهت رفع برخورد، از یک توزیع یکنواخت برای انتخاب قانون از میان قوانین مجموعه برخورد استفاده می‌نماید، بهره برده است و Player2، نیز نام یک رقیب کامپیوتری است.

در جداول ۲ و ۳، مقادیر پنج معیار امتیاز نهایی، منابع جمع‌آوری شده، ویژگی‌های اقتصادی (منابعی که در جهت بهبود اقتصاد مصرف شده)، امتیاز نظامی و تجربه دو بازیکن با یکدیگر مقایسه شده است.

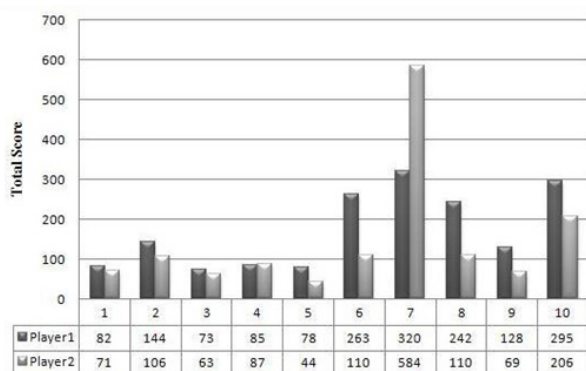
در جدول ۲ و در شکل ۹ (الف)، Player1 نام بازیکنی است که از سیستم تصمیم‌همیار هوشمندی که جهت رفع

جدول ۲- مقایسه امتیازات دو بازیکن در ده بار اجرای بازی

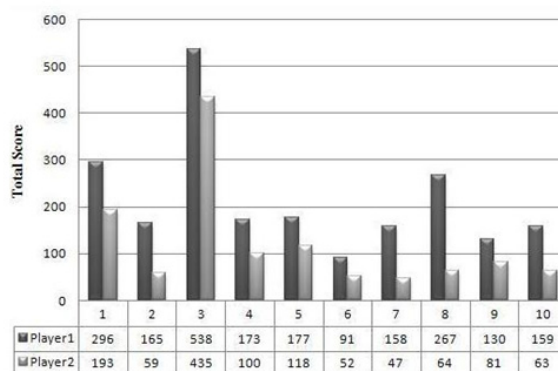
Player Name	Total Score	Resources	Economy (Spent on units + Spent on buildings + Spent on improvements)	Military	Experience	Winner
Player1	296	28413	10240+2200+2000	82	1879	✓
Player2	193	29605	16785+5150+1675	106	1203	
Player1	165	14420	7200+1450+0	74	8745	✓
Player2	59	9575	5400+1500+0	50	6753	
Player1	538	86996	49955+10622+4470	377	33078	✓
Player2	435	68883	29255+12000+7970	306	24932	
Player1	173	19527	11490+2300+1000	94	9349	✓
Player2	100	14161	6665+2600+100	78	7570	
Player1	177	18878	9450+1650+1350	84	8413	✓
Player2	118	19566	12600+4156+450	104	9354	
Player1	91	12426	8950+1450+0	68	9977	✓
Player2	52	7284	5000+950+0	66	6835	
Player1	158	14072	3200+2700+0	68	9915	✓
Player2	47	9160	3100+1000+100	40	6141	
Player1	267	28307	14180+5550+500	118	12726	✓
Player2	64	14064	15120+2955+300	114	8257	
Player1	130	12739	7150+1780+100	86	8592	✓
Player2	81	12843	8140+1650+785	76	7365	
Player1	159	17929	12540+1850+425	112	9563	✓
Player2	63	15476	9060+2095+600	88	8571	

جدول ۳- مقایسه امتیازات دو بازیکن دیگر در ده بار اجرای بازی

Player Name	Total Score	Resources	Economy (Spent on units + Spent on buildings + Spent on improvements)	Military	Experience	Winner
Player1	82	7306	2955+2011+0	24	5770	✓
Player2	71	7172	3660+1300+450	13	4381	
Player1	144	16330	8425+2750+150	67	8046	✓
Player2	106	12731	24445+1600+600	89	7487	
Player1	73	5229	1000+200+0	8	2859	✓
Player2	63	4049	2220+800+0	18	3804	
Player1	85	7534	4880+1050+0	59	4394	
Player2	87	7593	2260+700+200	34	6186	✓
Player1	78	6352	3300+875+0	19	4882	✓
Player2	44	4342	1000+470+0	6	2956	
Player1	263	29659	18180+3650+1325	140	15873	✓
Player2	110	26239	13075+5450+400	111	10991	
Player1	320	58206	30500+7459+7670	245	27503	
Player2	584	85194	40640+8715+18505	356	36312	✓
Player1	242	34956	16660+9444+0	146	16540	✓
Player2	110	21555	11575+5400+110-	102	13516	
Player1	128	15185	9640+1300+350	72	8556	✓
Player2	69	9840	5500+1865+0	48	7294	
Player1	295	39919	20635+6650+550	180	18825	✓
Player2	206	38656	19750+5251+2120	221	16751	



(ب)



(الف)

شکل ۹- نمودارهای امتیاز نهایی دو بازیکن. در هر دو نمودار Player2 نام یک رقیب کامپیوتری است. (الف) Player1 نام بازیکنی است که از سیستم تصمیم‌همیار هوشمندی که جهت رفع برخورد، از ایده تنوع نقطه نظرات استفاده می‌کند، کمک گرفته و (ب) Player1 نام بازیکنی است که از سیستم تصمیم‌همیار هوشمندی که جهت رفع برخورد، از یک توزیع یکنواخت برای انتخاب قانون از میان قوانین مجموعه برخورد استفاده می‌نماید، بهره برده است.

۵- نتیجه‌گیری و کارهای آینده

در این مقاله، سعی شد که با استفاده از ایده تنوع نقطه نظرات برای رفع برخورد در طی روند استنتاج، سیستمی به عنوان یک دستیار هوشمند برای یک بازی استراتژیک بلادرنگ، طراحی و پیاده‌سازی گردد. اعمال این ایده در چنین سیستمی، این امکان را برای تصمیم‌گیرنده فراهم

همان‌طور که دیده می‌شود، Player1 جدول ۲، نتایج بهتری از Player1 جدول ۳، که دو بار بازی را واگذار نموده، بدست آورده است.

از جمله کارهایی که در آینده روی این سیستم می‌توان انجام داد، کسب اطلاعات بیشتر از وضعیت سایر بازیکنان، تحلیل این اطلاعات و افزودن آن به سیستم در جهت افزایش دانش خاص سیستم از مسئله می‌باشد. همچنین می‌توان با مشاهده و ثبت نرخ افزایش منابع مختلف در حین بازی، تخمینی از این نرخ محاسبه و به دانش سیستم اضافه نمود، تا سیستم بتواند به‌روز رسانی مقادیر منابع را به صورت خودکار انجام دهد.

می‌سازد که با دید وسیع‌تری نسبت به شرایط محیط، و آگاهی از نتایج هر یک از انتخاب‌های ممکن، تصمیم‌گیری کند. اما به‌طور کلی، در استفاده از این ایده برای رفع برخورد در یک سیستم هوشمند، ضروری است به عاملی که منجر به توقف پیشروی درخت استنتاج، در هر یک از خطوط مجزای متناظر با قوانین ناسازگار، می‌گردد توجه کافی نمود. در واقع این عامل باید به گونه‌ای انتخاب گردد که در رابطه متوازن دوطرفه میان مدت زمان لازم برای به‌دست آوردن پاسخ نهایی، و تعداد مراحل موجود در پیشبرد استنتاج در خطوط مجزا، تعادل برقرار کند.

مراجع

- [1] Belavkin, R.V. (2003), "Conflict resolution by random estimated costs". 7th European Simulation Multi-conference (c) SCS Europe BVBA.
- [2] Simon, H.A. (1997), "Administrative Behavior: A Study of Decision-Making Process in Administrative Organizations". Free Press.
- [3] Shim, J.P., et al. (2002), "Past, present and future of decision support technology". Decision Support Systems, Vol. 33, pp. 111-126.
- [4] Gupta, J., et al. (2006), "Intelligent Decision-making Support Systems, Foundations, Applications and Challenges". Springer.
- [5] Turban, E., Aronson, J. (2001), "Decision Support Systems and Intelligent Systems". Prentice-Hall International.
- [6] Metoyer, R., et al. (2010), "Explaining how to play real-time strategy games". Knowledge-Based Systems, Vol. 23, pp. 295-301.
- [7] Hicks, R.C. (2007), "The no inference engine theory-performing conflict resolution during development". Decision Support Systems, Vol. 43, pp. 435-444.
- [8] Culbert, C., et al. (2007), CLIPS Reference Manual. International Thompson Publishing.
- [9] Anderson, J.R., Lebiere, C. (1998), "The Atomic Components of Thought". Mahwah, NJ: LEA.
- [10] Buchanan, B.G. (2005), "A very brief history of artificial intelligence". AI Magazine, Vol. 26, pp. 53-60.
- [11] Samuel, A.L. (1995), "Some studies in machine learning using the game of checkers". Computers & Thought, pp. 71-105.
- [12] Schaeffer, J. (2001), A gamut of games. AI Magazine, Vol. 22, pp. 29-46.
- [13] Hsieh, J.L., Sun, Ch.T. (2008), "Building a player strategy model by analyzing replays of real-time strategy games". International Joint Conference on Neural Networks (IJCNN).
- [14] Kirby, N. (2010), Introduction to Game AI. Course Technology PTR.
- [15] Proctor, M., et al. (2008), Drools Documentation. JBoss Community.
- [16] Buro, M., Furtak, T.M. (2004), "RTS games and real-time AI research". Behavior Representation in Modeling and Simulation Conference (BRIMS), Arlington, VA.
- [17] Ponsen, M., Spronck, P. (2004), Improving adaptive game AI with evolutionary learning, Computer Games: Artificial Intelligence, Design and Education, pp. 389-396.

- [18] Fasciano, M.J. (1996), "Everyday-world plan use". Technical Report TR-96-07, The University of Chicago, Computer Science Department, Chicago, Illinois.
- [19] Laird, J.E. (2001), "Using a computer game to develop advanced AI". Computer, Vol. 34, pp. 70-75.
- [20] Giarratano, J.C. (2003), CLIPS 6.21 User's Guide. International Thompson Publishing.
- [21] Wikipedia, the free encyclopedia. (Last visited in May 2011), "Age of Empires III", Available at: http://en.wikipedia.org/wiki/Age_of_Empires_III.

DESIGN AND IMPLEMENTATION OF AN IDSS CONFLICT RESOLVER USING DIFFERENT VIEW POINTS

F. Bonakdari^{1,*}, S. Jafari²

1. M.Sc., Computer Engineer, Shiraz University
2. Assistant Prof. of Computer Engineering, Shiraz University

*Corresponding Author: faeze.bonakdari@gmail.com

ARTICLE INFO

Keywords:

Conflict resolution,
Intelligent Decision
Support System,
Real-time strategic
game.

ABSTRACT

Conflict Resolution is an important procedure in many intelligent systems such as rule-based systems. This procedure determines the sequence of firing rules when more than one rule can fire. This paper presents an Intelligent Decision Support System wherein when a conflict occurs; it follows distinct reasoning line for each incompatible rule, and informs the decision maker about all possible options. This idea is applied in a system which is implemented for the purpose of assisting a player in a Real-Time Strategic Game. Obtained results indicate that player with help from this intelligent assistant can have better achievements over other players.
