

საქართველოს ტექნიკური უნივერსიტეტი

ხელნაწერის უფლებით

გიორგი აბელაშვილი

**მრავალაგენტიან ვიდეოთამაშებში კოლექტიური გადაწყვეტილებების
მოდელები**

დოქტორის აკადემიური ხარისხის მოსაპოვებლად წარდგენილი
დისერტაციის

ა ვ ტ ო რ ე ფ ე რ ა ტ ი

სადოქტორო პროგრამა „ინფორმატიკა“ შიფრი 0401

თბილისი

2016 წელი

სამუშაო შესრულებულია საქართველოს ტექნიკურ უნივერსიტეტში
ინფორმატიკისა და მართვის სისტემების ფაკულტეტი
ეკონომიკური ინფორმატიკის იდეპარტამენტი

ხელმძღვანელი : პროფ. ზურაბ ბოსიკაშვილი

რეცენზენტები-----

დაცვა შედგება-----წლის "-----"-----,----- საათზე
საქართველოს ტექნიკური უნივერსიტეტის-----
-----ფაკულტეტის სადისერტაციო საბჭოს კოლეგიის
სხდომაზე, კორპუსი ----- , აუდიტორია----- მისამართი:
0175, თბილისი, კოსტავას 77.

დისერტაციის გაცნობა შეიძლება სტუ-ს
ბიბლიოთეკაში, ხოლო ავტორეფერატის - ფაკულტეტის გვერდზე

სადისერტაციო საბჭოს მდივანი პროფ. თინათინ კაიშაური.

ნაშრომის ზოგადი დახასიათება

ვიდეოთამაშებში ხელოვნური ინტელექტის მრავალფეროვანი ამოცანები არსებობს. ძირითადი პრინციპი საერთოა - ხელოვნურმა აგენტმა რაც შეიძლება მეტი წინააღმდეგობა გაუწიონ ადამიან-მოთამაშეს, რათა მიზანი ნაკლებად იქნეს მიღწეული. რადგან თამაშები გათვლილია მომხმარებელზე, ასევე გათვალისწინებულია ხელოვნური ინტელექტისა და აგენტების დონეები. რაც უფრო მეტად "ჭკვიანია" აგენტი, მით უფრო რთულია მისთვის თამაშის მოგება.

მულტიაგენტური ტექნოლოგია ფართოდ გამოიყენება ვიდეოთამაშებში. იგი აერთიანებს თამაშების თეორიის, რთული სისტემების, გამოთვლითი სოციოლოგიისა და ხელოვნური ინტელექტის საკითხებს. ვიდეოთამაშებში ხელოვნური ინტელექტის ამოცანები, ძირითადად, დამყარებულია ინდივიდუალურ აგენტზე - ერთი ინდივიდი, რომელსაც არ ჰყავს არც დამხმარე და არც ალტერნატივა, ცდილობს მოთამაშეს გაუწიოს მეტოქეობა, გამოიცნოს მისი სვლები, დაიმახსოვროს გარკვეული გადაწყვეტილებები, გააკეთოს დასკვნები. ინდივიდუალური აგენტის შემთხვევაში ხელოვნური აგენტის მოქმედება უნიფიცირებულია და მორგებულია როგორც ერთ ინდივიდზე.

ვიდეოთამაშებში მულტიაგენტურ ტექნოლოგიის ძირითადი ხიბლი მდგომარეობს აგენტთა გაერთიანებებში საერთო ცოდნისა და მიზნის ქვეშ. წარმოვიდგინოთ სიტუაცია, როდესაც მოთამაშეს ეწინააღმდეგება არა ერთი აგენტი, როგორც ერთი ხელოვნური ინდივიდი, არამედ მსგავსი, ან სხვადასხვა ტიპის აგენტთა გაერთიანება. აგენტთა გაერთიანება შეიძლება იყოს მოქნილი მექანიზმი ხელოვნური ინტელექტის ამოცანისთვის.

დისერტაციის კვლევის საგანი ვიდეოთამაშების ხელოვნური ინტელექტის კოლექტიური გადაწყვეტილებების მოდელირება. ვიდეოთამაშებში აგენტთა ერთმანეთთან ცოდნის გაცვლა, რითაც საბოლოო ჯამში, მიიღება თამაშების ხელოვნური ინტელექტის კოლექტიური გადაწყვეტილებები.

იმისათვის, რომ კოლექტიური გადაწყვეტილებების მექანიზმი ავაგოთ, საჭიროა განვიხილოთ ერთაგენტიანი და მრავალაგენტიანი

სისტემები, მათი მოქმედების პრინციპები. მნიშვნელოვანია ცოდნის გადაცემის მექანიზმი, ნაშრომში ასევე გვაქვს მეცნიერულად განხილული არგუმენტაციის მექანიზმი.

კვლევის მიზანია შევქმნათ მრავალაგენტიან ვიდეოთამაშებში კოლექტიური გადაწყვეტილებების მიღების მექანიზმი. უამრავი თამაში არსებობს, სადაც ხელოვნური ინტელექტის მრავალფეროვანი ამოცანებია მოცემული. სამწუხაროდ, ბევრი ამოცანის წყარო დახურულია და მხოლოდ კონკრეტული მფლობელებისათვის, ან კომერციული კლიენტებისათვისაა წვდომადი. ჩვენ ამოცანას ჩამოვაყალიბებთ, როგორც ღია კოდის მქონე რესურსს, რომელსაც გამოიყენებენ სხვადასხვა ვიდეოთამაშების შემქმნელები.

სადისერტაციო ნაშრომში მეცნიერულ სიახლეს წარმოადგენს:

- მრავალაგენტიან სისტემაში ცოდნის გადაცემის ტიპი – ცოდნის მარკირება, როდესაც ჩვენ შეგვიძლია მრავალაგენტიან სისტემაში რაღაც ნიშნით, გარკვეულ ადგილას დავტოვოთ ცოდნა სხვა ინტელექტუალური აგენტისთვის. მისთვის ეს ცოდნა იქნება პირდაპირი გზით აღებული, რაც სისტემის მუშაობის სისწრაფეს განსაზღვრავს.
- მრავალაგენტიან ვიდეოთამაშებში, ცოდნის მარკირების ამოცანით და არგუმენტაციის კონკრეტული მექანიზმით კოლექტიური გადაწყვეტილებების მექანიზმის შექმნა. ამ მექანიზმის მიხედვით ხდება აგენტთა ჯგუფების დინამიური ფორმირება და კოლექტიურად მოქმედება.

ჩვენს ნაშრომში ჩატარებული კვლევა, შეიძლება გახდეს უნიფიცირებული ფრეიმვორქი, რომლის გამოყენებაც შეიძლება თამაშების სფეროს გასცდეს. შეიძლება ჩვენი სისტემა მოერგოს ზოგად სფეროებს, სადაც არგუმენტაციის მექანიზმი, ცოდნის გადაცემა და მრავალაგენტიანი მექანიზმების კოლექტიური გადაწყვეტილებები აქტუალურია.

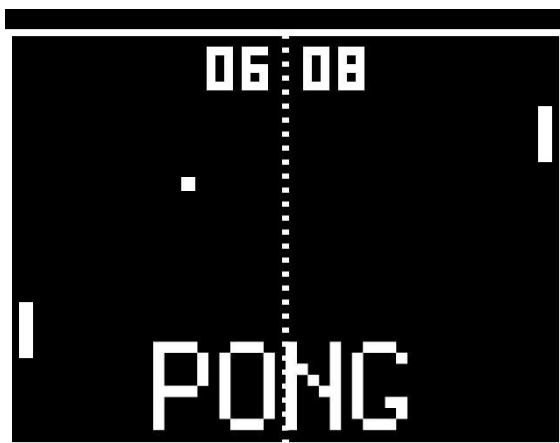
დისერტაცია, რომელსაც წარმოადგენს ეს ავტორეფერატი, შედგება 126(ასოცდაექვსი) გვერდისაგან და მოიცავს ხელოვნური ინტელექტის ზოგად მიმოხილვას ვიდეოთამაშებში, ცოდნის წარმოდგენას და დამუშავებას ვიდეოთამაშებში, აღწერილია კონკრეტული ამოცანა-

ვიდეოთამაში - "ჭიანჭველების ამოცანა", სადაც ჩვენი საკვლევი თემის ექსპერიმენტი აღწერილია. აღწერილია სიმბოლოთა გამოცნობის მექანიზმები, ასევე არგუმენტაციის მექანიზმი ვიდეოთამაშებში და არგუმენტაციის საფუძველზე კოლექტიური გადაწყვეტილებების მექანიზმი ვიდეოთამაშებში.

ხელოვნური ინტელექტი ვიდეოთამაშებში, ძირითადად, არის გადაწყვეტილებათა მიმღები სისტემა. ამ სისტემით მოქმედებენ არამოთამაშე სუბიექტები და ობიექტები ადრეული თამაშები მაგ. Pac Man(სურ1), Pong(სურ 2). წარმოადგენდნენ მარტივი ხელოვნური ინტელექტის ნიმუშებს. ამ თამაშებშიარსებული გარემო არ საჭიროებდა განსაკუთრებულ ალგორითმების მოფიქრებას. თუმცა თანამედროვე თამაშებმა წარმოშვა ისეთი სირთულეები, როგორცაა რეალური სამყაროს მოდელის დამოკიდებულება მოთამაშესა და გარემოზე Assassin's Creed(სურ 3), Farcry 3(სურ 4). თანამედროვე თამაშებში გარემო იყო მაქსიმალურად სრულყოფილი და მიახლოებული რეალურთან.



სურ. 1 Pac Man



სურ. 2 Pong



სურ. 3 Assassin's Creed



სურ. 4 Farcry 3

ხელოვნური ინტელექტის ძირითადი მიზანია შექმნას მოთამაშისთვის საინტერესო შეჯიბრის გარემო. მოთამაშისთვის ერთადერთი მიზანი კი თამაშიდან მიღებული სიამოვნებაა. დავახასიათოთ თამაშების ხელოვნური ინტელექტის 7 ძირითად მიზანი, ეს მიზნები განსაზღვრავს ინტელექტის სირთულესაც.

1) **სამართლიანი თამაში** - ხელოვნური ინტელექტი უნდა ემორჩილებოდეს თამაშის ყველა წესს, არ უნდა იქცეოდეს ამ წესების გვერდის ავლით, რათა მოთამაშისთვის უცნობი გარემო არ შეიქმნას.

2) **ამოუცნობი ქცევა** - ხელოვნური ინტელექტის ნებისმიერი ქცევა უნდა იყოს მოთამაშისთვის მოულოდნელი. რატომღაც უინტერესო

იქნებოდა თამაში წინასწარ რომ ვიცოდეთ ჩვენი მოწინააღმდეგე რას მოიმოქმედებს.

3) არაცხადი ქცევა - სისტემის ნებისმიერ შემდეგი ქცევა არ უნდა იყოს აშკარა მოთამაშისთვის, არ უნდა აფროთხილებდეს მას.

4) გარემოს გამოყენება - სისტემა ნებისმიერ გარემოს თვისებას უნდა იყენებდეს ინტელექტუალურად მის სასარგებლოდ. საინტერესო და რთული სათამაშო იქნებოდა თუ მტერი ნისლიან გარემოში დაემალებოდა მოთამაშეს.

5) თვითშესწორება - ხელოვნური ინტელექტს უნდა შეეძლოს საკუთარი ქცევების კორექტირება და ცოდნის რედაქტირება. ნებისმიერ დროს უნდა შეეძლოს შაბლონის გადაკეთება და სხვა სტილით თამაში, ამ შემთხვევაში მოთამაშე რთულად აუღებს ალღოს მის პრინციპებს.

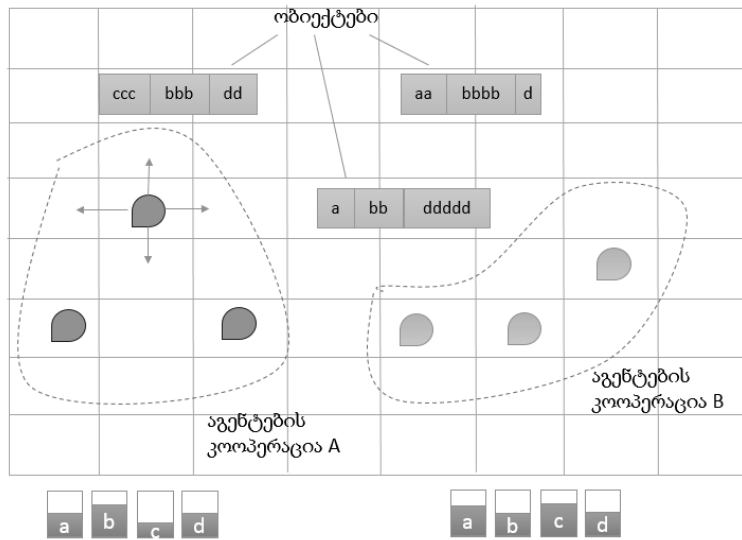
6) კრეატიულობა - ხელოვნური ინტელექტი უნდა პოულობდეს გამოსავალს ნებისმიერი სიტუაციიდან, მათ შორის გაუთვალისწინებელიდანაც.

7) ადამიანის მსგავსი ქცევა - ხელოვნური აგენტის ქცევა მაქსიმალურად უნდა იყოს მიახლოებული მოთამაშის ქცევასთან.

დისერტაციის კვლევის საგანი, მულტიაგენტური გაერთიანებები განვიხილოთ კონკრეტულ მაგალითზე. ამისათვის, საჭიროა შევქმნათ ისეთი ვიდეოთამაში, სადაც მონაწილეობას მიიღებენ ინტელექტურლური აგენტები. აგენტებს ექნებათ ცოდნის ბაზა, გადაწყვეტილებების გამოტანის მექანიზმი და შესაბამისად იმოქმედებენ.

ნებისმიერ ვიდეოთამაშს აქვს ერთი ძირითადი პრინციპი, იგი არის დელიბრილებული, ანუ გვაქვს ჩაკეტილი სამყარო, სადაც შეგვიძლია მოქმედება. ჩვენი ამოცანაც დავაფუძვნოთ ამ პრინციპზე.

მოცემულია ორგანზომილებიანი სიბრტყე, სიბრტყეზე განლაგებულნი არიან ჭიანჭველა აგენტები. მათ შეუძლიათ ამ სიბრტყეზე მოძრაობა მხოლოდ ვეერტიკალურად, ან ჰორიზონტალურად.



სურ. 5 ჭიანჭველების თამაშის გრაფიკული პროტოტიპი

სიბრტყეზე შემთხვევითად ჩნდებიან ობიექტები(ფოთლები). ობიექტები წარმოადგენენ შეზღუდული ანბანის შედგენილ მიმდევრობათა მიმდევრობას.(სურ.5)

ჭიანჭველა აგენტებს შეუძლიათ შემდეგი ტიპის მოქმედებები.

- შეზღუდული ზომის ფოთლის გადაადგილება
- ფოთლის დაჭრა იმ სიმბოლოზე რომელსაც იცნობს
- გამოყოფილი სიმბოლოს შენახვა საცავში

თამაშის მიზანია გარკვეულ დროში, კოოპერაციაში მყოფმა ჭიანჭველა აგენტებმა რაც შეიძლება მეტისიმბოლოები დააგროვონ საცავში.

აგენტები, თავდაპირველად, სიბრტყეზე ჩნდებიან შემთხვევით პოზიციებზე. მათ შეუძლიათ როგორც ინდივიდუალური მოქმედება, ასევე კოოპერაციის შექმნა და კოპერაციული გადაწყვეტილებები. კვლევის მიზანიც სწორედ ამაში მდგომარეობს, შევქმნათ აგენტთა ჯგუფური გადაწყვეტილებების მოდელი.

ჭიანჭველების ამოცანის იმპლემენტაცია, როგორც თამაში, რამდენიმენაირად შეიძლება. ძირითადი განსხვავება იქნება მოქმედ ინტელექტუალურ აგენტებში - ისინი შეიძლება ამოცანას სხნიდნენ დამოუკიდებლად, ასევე კოოპერაციულად. განვიხილოთ თვითოეული თამაშის ასეთი განსხვავებული სცენარი.

ინდივიდუალური აგენტების შემთხვევაში, ყოველ ჭიანჭველას გააჩნია რანდენიმე მოქმედების ვარიანტი:

- მოძრაობა, ვერტიკალურად, ან ჰორიზონტალურად.

- გააჩნია რადიუსი, რითაც იგი პოულობს რაიმე ობიექტს.
- ობიექტის გარჩევის უნარი, არის თუ არა ის ფოთოლი.
- ფოთლის ვარგისიანობის დადგენა - ვარგისია თუ არა წასადებად
- თვითოეული ჭიანჭველა სცნობს გარკვეულ სიმბოლოს
- მოაჭრას შესაბამისი სიმბოლო თუ მოიპოვება იგი ფოთოლში
- გადაიტანოს სიმბოლო საცავში და დაბრუნდეს უკან.

ერთი შეხედვით, ჩვენ წარმოგვიდგა რთული მოდელი ინდივიდუალური აგენტისათვის. სიტყვაში რთული, იგულისხმება აგენტის მოქმედების მასშტაბურობა, მისი ცოდნის ბაზის მრავალფეროვნება.

თვითოეული ჭიანჭველა თამაშს იწყებს საწყისი ცოდნით, ცოდნის ბაზაში გაწერილია წესები, რითაც იგი არჩევს გარკვეულ ობიექტს ფოთლისგან, ასევე, იცის რას ნიშნავს ვარგისიანი ფოთოლი და რაც მთავარია, სცნობს თავის სიმბოლოს. განვიხილოთ ერთაგენტიანი ჭიანჭველათა ამოცანის სიმულაციის ერთი ნაბიჯი.

ჭიანჭველა ჩნდება შემთხვევითად გარკვეულ ადგილას სიბრტყეზე, ის იწყებს მოძრაობას ასევე შემთხვევითად. მას, როგორც ვთქვით გააჩნია, ობიექტის პოვნის გარკვეული რადიუსი. რაიმე ობიექტის პოვნის შემთხვევაში ჭიანჭველა მიმართავს ცოდნის ბაზას, შეადარებს ობიექტს ფოთლის პარამეტრებს და გამოიტანს გადაწყვეტილებას, ეს ობიექტი არის თუ არა ფოთოლი. უარყოფითი პასუხის შემთხვევაში, გააგრძელებს მოძრაობას. ხოლო დადებითი პასუხის შემთხვევაში გადადის შემდეგ ნაბიჯზე. ამ დროს ჭიანჭველამ ამ ოპერაციისთვის დახარჯა $O(n)$ დრო, n - ამოცნობისათვის საჭირო ოპერაციების რაოდენობა.

ფოთლის პოვნის შემდეგი ნაბიჯია, მისი ვარგისიანობის დადგენა, შეიცავს თუ არა შესაბამის სიმბოლოს. სიმბოლოს შემოწმება ხდება ფოთლის ყველა შემავალი სიმბოლოს გადამოწმებით - თუ ფოთოლში ვერ მოიძებნა ასეთი ასო, ჭიანჭველა მიატოვებს ფოთოლს და წავა სხვა ფოთლის საძებნელად, პოვნის შემთხვევაში გადადის დაჭრისა და ტრანსპორტირების ეტაპზე.

საბოლოო ეტაპი ფოთლის დამუშავებისა არის სიმბოლოს მოჭრა და გადატანა საცავში. როდესაც ჭიანჭველა მოჭრის ნაპოვნ ასოს, იგი მაშინვე წაიღებს საცავში ამ სიმბოლოს, დაიმასსოვრებს ბოლო კოორდინატს სადაც ვარგისი ფოთოლი იპოვა, რათა ნაჭრის გადატანის შემდეგ დაბრუნდეს და გააგრძელოს ფოთლის დამუშავება.

დამასსოვრებულ კოორდინატზე დაბრუნების შემდეგ ჭიანჭველა თავიდან იმეორებს ფოთლის დამუშავების ბიჯებს, ეძებს სიმბოლოს, პოვნის შემთხვევაში იგი მოაჭრის და გადაიტანს და კვლავ დაბრუნდება. ასე ივლის რეკურსიულად მანა, სანამ ფოთოლში არ ამოიწურება შესაბამისი სიმბოლოები.

განხილული მაგალითის მიხედვით, ინდივიდუალურ აგენტზე დაფუძნებული ქცევებით, ჭიანჭველების ამოცანა ამოიხსნება, თუმცა არ იქნება ეფექტური. რა ოპტიმიზაციაც არ უნდა გავუკეთოთ ტექნიკურ ალგორითმებს, იდეა მაინც შეანელებს პროცესს. ჭიანჭველას ძალიან ბევრჯერ უხდება ერთი და იგივე ხანგრძლივი ბიჯების გამეორება. იგი არც ცოდნას ინახავს, არც გადასცემს არავის და შესაბამისად, არც არავისგან იღებს. კარგი იქნებოდა ნებისმიერი ფოთლისთვის მას პირდაპირ მიეღო ინფორმაცია შესაბამისი სიმბოლოს არსებობის, ან არ არსებობის შესახებ, რათა მას დაეზოგა სიმბოლოთა შემოწმების ოპერაცია, ასევე კარგი იქნებოდა დაჭრისას არა მხოლოდ ერთი არამედ რამდენიმე სიმბოლო გამოეცალკეებინა, ისევე როგორც გადაეტანა რამდენიმე სიმბოლო.

ზემოთ ვახსენეთ ცოდნის გადაცემა ინტელექტუალურ აგენტებში, რასაც არ იყენებს ჭიანჭველათა ამოცანის ინდივიდუალური აგენტის მოდელი. იმისათვის, რომ ეს ამოცანა უფრო მოქნილი და დროში აჩქარებული გავხადოთ, საჭიროა შემოვიღოთ მულტიაგენტური ქცევები, მულტიაგენტური ცოდნა და ცოდნის გადაცემა. სანამ ჭიანჭველათა მაგალითს განვიხილავთ მულტიაგენტური მხრიდან, განვიხილოთ ცოდნის გადაცემის მექანიზმი ვიდეოთამაშებში.

ზემოთგანხილული საკითხებიდან დავასკვნით, რომ ზოგიერთ ამოცანაში საჭიროა გვექონდეს მულტიაგენტური მექანიკა. ჩვენ უკვე ვიცით რას ნიშნავს მულტიაგენტურ ვიდეოთამაშში ცოდნის გადაცემა, ვიცით რა უპირატესობა აქვს ინტელექტუალურ აგენტთა როლებად და

კლასებად დაყოფას. ჭიანჭველების ამოცანა განვიხილოთ როგორც მულტიაგენტური ვიდეოთამაში, სადაც ძირითადი წარმადობის მექანიკა მდგომარეობს ცოდნის მარკირებით გადაცემაში.

ჭიანჭველა მზვერავი მოძრაობს ველზე ნებისმიერი მიმართულებით. მისი რადიუსისდა შესაბამისად პოულობს ობიექტებს და იწეებს მათ გარჩევას ფოთლისგან. ხდება ცოდნის ბაზაში ფაქტებისა და პირობების გადამოწმება. განვიხილოთ ერთი სიმულაციური ბიჭვი მზვერავი ჭიანჭველის მოქმედებისა, როდესაც ის პოულობს ფოთოლს. მზვერავის ცოდნის ბაზა დაახლოებით ასე გამოიყურება:

ფაქტები:

ფაქტი1: ფოთოლი მწვანეა

ფაქტი2: ფოთოლი მსუბუქია 0.01 გრამზე

პირობები:

პირობა1:თუ ნაპოვნი ობიექტი მწვანეა, შეამოწმე ახალი პირობა მეორე ფაქტზე

თუ არა, დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე იმის შესახებ, რომ ეს ობიექტი არ არის ფოთოლი. გააგრძელე ძებნა.

პირობა2: თუ ნაპოვნი ობიექტი 0.01 გრამზე მსუბუქია, ის ფოთოლია, დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე რომ ის ფოთოლია. თუ არა - დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე იმის შესახებ, რომ ეს ობიექტი არ არის ფოთოლი და გააგრძელე ძებნა.

მზვერავის მოქმედების თანმიმდევრობა მკაცრადაა განსაზღვრული, მან თუ იპოვა ნებისმიერი ობიექტი იგი თავის ცოდნის ბაზაზე დაფუძნებით იწეებს ანალიზს, არის თუ არა იგი ფოთოლი. გარკვევის შემთხვევაში იგი ტოვებს ცოდნას კონკრეტულ ადგილას, რომ ფოთოლი უპოვია, ხოლო არ პოვნის შემთხვევაში შესაბამისად იქცევა, მხოლოდ არგუმენტი არის უცნობი ობიექტი, ანუ არ უპოვია ფოთოლი.

ცოდნის დატოვების ქცევით შესაბამისი მზვერავი უმარტივეს საქმეს უკვე შემდეგ მომავალ მზვერავს, რომელიც კონკრეტული მიზნით მოადგება ამავე ობიექტს, მას შეუძლია მხოლოდ შეამოწმოს მისი წინამორბედის დატოვებული ნიშანი, არის თუ არა ფოთოლი ეს

ობიექტი. დადებითი და უარყოფითი პასუხის შემთხვევაში აღარაფერს შეამოწმებს, რადგან მან უკვე იცის ინფორმაცია ამ ობიექტზე, გააგრძელებს გზას. მზვერავის მოქმედების ფსევდო აღწერა შემდეგნაირად გადაკეთდება:

პირობა 1ა. თუ ნაპოვნ ობიექტთან უკვე არის მარკირებული ცოდნა მის რაობაზე, გააგრძელებს გზას, თუ არა და გადადი პირობა 1-ზე.

პირობა1: თუ ნაპოვნი ობიექტი მწვანეა, შეამოწმე ახალი პირობა მეორე ფაქტზე

თუ არა, დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე იმის შესახებ, რომ ეს ობიექტი არ არის ფოთოლი. გააგრძელებს ძებნას.

პირობა2: თუ ნაპოვნი ობიექტი 0.01 გრამზე მსუბუქია, ის ფოთოლია, დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე რომ ის ფოთოლია. თუ არა - დატოვე ცოდნა კონკრეტულ ადგილას კონკრეტულ ობიექტზე იმის შესახებ, რომ ეს ობიექტი არ არის ფოთოლი და გააგრძელებს ძებნას.

როგორც ვხედავთ, ცოდნის მარკირების გამოყენებით ჭიანჭველა მზვერავს ძალიან შეიძლება გაუმარტივდეს საქმე. იგი ამ მექანიზმის გამოყენებით დაზოგავს მთელ რიგ ოპერაციებს, რომელიც კავშირშია ობიექტის ანალიზთან ცოდნის ბაზაზე დაყრდნობით, ხოლო ცოდნის მარკირებით, მას პირდაპირ მიეწოდა დამუშავებული ინფორმაცია.

რაც შეეხება ჭიანჭველა ფოთლის მჭრელებს, მზვერავების მსგავსადაც ორგანიზმილებაში მოქმედებენ. მათი ფუნქციაა ფოთლის დაჭრა, თუმცა რა იციან, რომ ობიექტი ფოთოლია. მათი ამ ცოდნის შესავსებათ გვაქვს ორი ვარიანტი:

1) ობიექტის პოვნის შემთხვევაში გააკეთონ მისი ანალიზი მზვერავის შესაბამისად

2) კონკრეტულ ობიექტთან მიმართებაში იხელმძღვანელონ მზვერავის მიერ მარკირებული ინფორმაციით.

ცხადია, პირველი ვარიანტი არ გამოდგება. ჩვენ ვთქვით, რომ განვიხილავთ ჭიანჭველების ამოცანის მულტიაგენტურ ვარიანტს. ჭიანჭველები დაყვავით კლასებად, თვითოეულს გააჩნია თავისი როლი და მთავარი ფუნქცია. ფოთლის მჭრელს უბრალოდ, არ შეუძლია

ფოთლის ამოცნობა, იგი მხოლოდ სიმბოლოებს სცნობს. გვრჩება მეორე ვარიანტი, რომელიც საუკეთესო გამოსავალია - ფოთლის მჭრელი მოქმედებს მზვერავის ინფორმაციაზე დაყრდნობით. განვიხილოთ მზვერავის ცოდნის ბაზა, ფაქტები და პირობები:

ფაქტი 1 : ვცნობ მზვერავის დატოვებულ ინფორმაციას(ობიექტი ფოთლია თუ არ არის ფოთლი)

ფაქტი 2 : სიმბოლო, რომლის ამოცნობაც შემძლია გამოიყურება როგორც T

პირობები:

პირობა 1: თუ ობიექტი მარკირებულია ცოდნით გადადი პირობა 2-ზე, თუ არა, გააგრძელე გზა სხვა ფოთლის ძიებაში.

პირობა 2: თუ მარკირებული ობიექტი ფოთლია გადადი პირობა 3-ზე, თუ არა გააგრძელე გზა სხვა ფოთლის ძიებაში.

პირობა 3: შემოწმე, თუ ფოთლი შეიცავს სიმბოლო T, მოაჭერი სიმბოლო და დატოვე ინფორმაცია ამის შესახებ, კონკრეტულად, დარჩა თუ არა შესაბამისი სიმბოლო ისევ ფოთლში, სულ რამდენი სიმბოლო არის მოჭრილი. თუ არ შეიცავს ფოთლი სიმბოლოს, დატოვე ინფორმაცია ამის შესახებ, რომ არ შეიცავს. გააგრძელე გზა.

ფოთლის მჭრელიც, შეიძლება ითქვას, დამოკიდებულია მზვერავის შენახულ ინფორმაციაზე. ობიექტის პოვნისას, იგი იმ შემთხვევაში გადადის მოქმედებაზე, თუ მან მზვერავის დატოვებული ცოდნა აღმოაჩინა კონკრეტულ ობიექტზე, თუ ვერ აღმოაჩინა აგრძელებს გზას. საინტერესოა ისიც, რომ სიმბოლოს მოჭრის თუ არ მოჭრის შემთხვევაში, მჭრელი ტოვებს ცოდნას მის კონკრეტულ მოქმედებაზე. ეს ცოდნა სხვა ფოთლის მჭრელს გამოადგება, მას შეუძლია პირდაპირ გაიგოს არის თუ არა მისი შესაბამისი სიმბოლო ამ ფოთლში. ამ ფაქტზე დაფუძნებით, ფოთლის მჭრელის ქცევა იცვლება შემდეგნაირად:

პირობა 1: თუ ობიექტი მარკირებულია ცოდნით გადადი პირობა 2-ზე, თუ არა, გააგრძელე გზა სხვა ფოთლის ძიებაში.

პირობა 2: თუ მარკირებული ობიექტი ფოთლია, გადადი პირობა 2.1-ზე, თუ არა გააგრძელე გზა სხვა ფოთლის ძიებაში.

პირობა 2.1 : შეამოწმე, თუ არსებობს წინა მჭრელების ინფორმაცია საკუთარ სიმბოლოზე, ფოთოლში არსებობს სიმბოლო? გადადი პირობა 3-ზე, არ არსებობს? გააგრძელე გზა.

პირობა 3: შეამოწმე, თუ ფოთოლი შეიცავს სიმბოლო T, მოაჭერი სიმბოლო და დატოვე ინფორმაცია ამის შესახებ, კონკრეტულად, დარჩა თუ არა შესაბამისი სიმბოლო ისევე ფოთოლში. თუ არ შეიცავს, დატოვე ინფორმაცია ამის შესახებ, რომ არ შეიცავს. გააგრძელე გზა.

წინა კოლების ცოდნაზე დაფუძნებით, მას შეუძლია დაზოგოს სიმბოლოს შემოწმების ოპერაციები. იგი პირდაპირი გზით მიიღებს შეტყობინებას თავისი სიმბოლოს ფოთოლში არსებობის შესახებ.

ჭიანჭველა-გადამზიდი აგენტის ფუნქცია მარტივია - იპოვოს მოჭრილი ფოთლის ნაჭერი და წაიღოს საცავში, თუმცა მის ქცევაში საჭიროა გარკვეული ოპტიმიზაცია. განვიხილოთ ამ როლის აგენტის ცოდნა:

ფაქტი 1: ნებისმიერი მოჭრილი ფოთლის ზომა არ აღემატებაD-ს.

ფაქტი 1.1: მოჭრილი ფოთლის ნაჭერი შედგება ერთი სიმბოლოსგან.

ფაქტი 2: ყველა ფოთლის ნაჭერს, რომელსაც აქვს ერთი სიმბოლო, შეიძლება გადაიტანოს საცავში.

ფაქტი 3: შეუძლია გადაიტანოს ფოთლის ნაჭერი(ნაჭრები) თუ ჯამში იწონის(იწონიან)m-მასას.

პირობები:

პირობა 1:თუ ნაპოვნი ობიექტის ზომა ნაკლებიაD-ზე. ის ფოთლის ნაჭერია, გადადი პირობა 2-ზე, თუ არა, გააგრძელე გზა სხვა ობიექტის ძიებაში.

პირობა 2: თუ ფოთლის ნაჭერი შეესაბამება წადების პირობას, ნუ ის მსუბუქია, გადავიდეს პირობა 3-ზე, თუ არა და წაიღოს ერთი ნაჭერი საცავში და ნაჭრის დაბინავების შემდეგ დაბრუნდეს, ან არ დაბრუნდეს ამ ადგილას.

პირობა 3: მოეცმულ ფოთლის მიდამოში შეამოწმოს ყველა არსებული ფოთლის ნაჭრებიდან რამდენის წადება შეუძლია.

მე-2 და მე-3 პირობები კითხვის ნიშანს ჰბადებს. რას ნიშნავს უკან დაბრუნება, ან რატომ უნდა დაბრუნდეს, ან არ უნდა დაბრუნდეს

გადამზიდისამუშაო ადგილას? ამ ქცევას განაპირობებს მისი წინა ქცევა - დარჩა კი წასაღები სიმბოლოები? თუ დარჩა, უნდა დაბრუნდეს მათ წასაღებად, ხოლო თუ ბოლო ნაჭერი წაიღო საცავში, მისი დაბინავების შემდეგ უნდაგააგრძელოს ძებნა სხვა ნაჭრებისთვის. შეიძლება ვიფიქროთ, რომ ნებისმიერ, უკან დაბრუნებულ ჭიანჭველა გადამზიდს დარჩენილი ნაჭრები არ დახვდეს, ანუ იგი კონკურენციის მსხვერპლი გახდეს, თუმცა ამ პრობლემის ოპტიმიზაცია შეიძლება. თუ ნებისმიერი ჭიანჭველა-გადამზიდი დატოვებს ცოდნას იმის შესახებ, დარჩა თუ არა სიმბოლოები ფოთლის მიდამოში, ნებისმიერ შემდეგ მომსვლელს პირდაპირი გზით გადაეცემა ეს ცოდნა. შესაბამისად, ისინი უკვე გააგრძელებენ სხვა ნაჭრების ძებნას და არა არსებულ ზონაში დარჩენილი ნაჭრების მოძებნას.

მე-3 პირობას რაც შეეხება, იმისათვის, რომ დავაოპტიმიზიროთ პირობის ფუნქცია, საჭიროა გადამზიდს გარკვეული სახით დახვდეს ცოდნა უკვე მოჭრილი ნაჭრების რაოდენობის შესახებ. იგი პირდაპირ მიმართავს შემოწმების ფუნქციას კონკრეტული რაოდენობით და არ დაიწყებს ხელახალ ძებნას. ამ ცოდნას კი, რასაკვირველია, პირდაპირი გზით მიიღებს წინამორბედი ფოთლის მჭრელიდან. როგორც ვახსენეთ, ნებისმიერი ფოთლის მჭრელი ტოვებს ინფორმაციას კონკრეტულ ადგილას მოჭრილი სიმბოლოების შესახებ.

მრავალაგენტოვანი ხელოვნიერი ინტელექტის ერთერთი ძირითადი მექანიკა მდგომარეობს კოლექტიურ გადაწყვეტილებებში. აგენტები ურთიერთქმედებენ ერთმანეთთან და ახდენენ საკითხის საერთო შემუშავებას. ასეთი მიდგომის გამოყენება ყოველთვის არ არის მიზანშეწონილი, რათქმაუნდა მოცანის ტიპზეა დამოკიდებული. ვიდეოთამაშებში უანრი და სცენარი მკაცრად განსაზღვრავს ტექნიკურ მახასიათებლებს. მაგალითად, ჭიანჭველების ამოცანა იდეალური ვარიანტია, რომ განვახორციელოთ კოლექტიური გადაწყვეტილებების სისტემა.

ზოგ გაერთიანებას სჭირდება MAS(MultyAgent System). უფრო დეტალურად, თუ ამ გაერთიანებაში არიან სხვადასხვა ორგანიზაციები, სხვადასხვა ტიპისა დამიზნების მქონე ხალხი, საჭიროა მრავალაგენტოვანი სისტემა მათი ურთიერთობისათვის. იმ შემთხვევაშიც

კი, თუ თითოეულ ორგანიზაციას სურს საკუთარი შინაგანი უშიშროების მოდელირება ერთი საერთო სისტემით, ორგანიზაციები არ მისცემენ ამ სისტემის შექმნის უფლებას მხოლოდ ერთ ადამიანს. სხვადასხვა ორგანიზაციებს სჭირდებათ საკუთარი სისტემა, რომელიც ყველაზე უკეთ აკმაყოფილებს მათ მოთხოვნებს. მაგალითისათვის, წარმოვიდგინოთ კომპანია X, რომელიც აწარმოებს საბურავებს, მაგრამ ხელშეკრულებით თანამშრომლობს თხილის პროდუქციაზე კომპანია თან. ამ პროდუქციის პროცესის ერთი სისტემაში მოსაყვანად, უნდა მოხდეს X-სა და -ის შიდა საქმეების მოდელირება. თუმცა არცერთ კომპანიას არ აწყობს საკუთარი ინფორმაციის ან კონტროლის სხვა კომპანიისათვის გასაჯაროება ან გადაცემა. ორი კომპანიის შემთხვევაში შესაძლოა მიიღწიოს შეთანხმებას, მაგრამ მრავალი კომპანიის შემთხვევაში MAS მნიშვნელოვანია.

სხვა მაგალითი შეიძლება იყოს საავადმყოფოს განრიგის შედგენა. ეს გარემო მოითხოვს სხვადასხვა აგენტებს რომ წარმოადგინოს სხვადასხვა ადამიანების ინტერესები საავადმყოფოში (სხვადასხვა გამოცდილებისა და პროფესიის ექიმები, რომლებსაც გააჩნიათ განსხვავებული განრიგი, ექთნები, ასევე ოპერატორები და ა.შ). რადგანაც ამ ადამიანებს გააჩნიათ სხვადასხვა განრიგი, სხვადასხვა კრიტერიუმების გათვალისწინებით, მათი წარმოდგენა საჭიროა სხვადასხვა აგენტებით.

იმ გარემოებებშიც, სადაც არ გამოიყენება დისტრიბუციულის სისტემები, ზოგჯერ საჭიროა MAS. მრავალი აგენტის არსებობამ შესაძლებელია ოპტიმიზაცია გაუკეთოს სისტემის წარმადობას, პარალელური გამოთვლის მეთოდის წარმოებით. MAS ის პარალელიზმი შესაძლოა დაეხმაროს დროით შეზღუდული ან სივრცით შეზღუდული მოთხოვნების დაკმაყოფილებას.

გარდა იმისა, რომ დავალებების სხვადასხვა აგენტებზე განაწილებით მიიღწევა პარალელიზმი, სიმძლავრე არის MAS ისუპირატესობა, რომელსაც ყავს ზედმეტი აგენტები. თუ კონტროლი და შესაძლებლობები თანაბრადაა გადანაწილებული სხვადასხვა აგენტებზე, სისტემის მუშაობა ბევრად სტაბილური და ეფექტურია. თუ

ერთი აგენტი (პროცესორი) მართავს ყველაფერს, ერთიუმნიშვნელო შეცდომა ხშირ შემთხვევაში განაპირობებს სისტემის ავარიულად დასრულებას. თუმცა არ არის აუცილებელი მრავალაგენტიანი სისტემა განაწილებული იყოს სხვადასხვა პროცესორებზე, სისტემის ავარიული მარცხის თავის ასარიდებლად აგენტები უნდა იყვნენ გადანაწილებული სხვადასხვა მანქანებზე.

მრავალაგენტიანი სისტემების კიდევ ერთი უპირატესობაა მასშტაბურობა. ისინი არსებითად მოდულურები არიან და კარგ მოდელირებულ სისტემაში ადვილია ახალი აგენტების დამატება რათა სისტემა გაფართოვდეს ახალი შესაძლებლობებით. ასევე ეს უპირატესობა მოიცავს არსებული მოდულების შესაძლებლობებისა და კონტროლის ცვლილების სიმარტივესა და მოქნილობას.

პროგრამისტის თვალსაზრისით, მოდულურობას მიყვავართ მარტივ პროგრამირებასთან. ერთი რთული ამოცანის ცენტრალიზებულ აგენტზე დავალების ნაცვლად, მას შეუძლია განსაზღვროს ქვე-ამოცანები და დაუწესოს კონტროლი სხვადასხვა აგენტებს ამ დავალებების შესასრულებლად.

მრავალაგენტიანი სისტემა, ინტელექტთან ერთად ასევე შეიძლება გამოყენებულ იქნას სოციალური და ცხოვრებისეული მეცნიერების ფუნდამენტური პრობლემის გადასაჭრელად. ინტელექტი ღრმად დაკავშირებულია ურთიერთქმედებასთან. ითვლება, რომ ყველაზე საუკეთესო გზის დასაწყისი ინტელექტუალური მანქანების შესაქმნელად არის სოციალური მანქანების შექმნა. ეს თეორია ეფუძნება სოციალურ- ბიოლოგიურ თეორიას, რომელიც მიიჩნევს, რომ ხელოვნური ინტელექტი სწორედ სოციალური ურთიერთობის საფუძველზე შეიქმნა.

ქვემოთ ჩამოთვლილია მიზეზები, თუ რატომ უნდა გამოვიყენოთ

MAS: მრავალაგენტიანი სისტემების გამოყენების მიზეზები

- რაღაც გარემოებები ითხოვენ მას
- პარალელიზმი
- სიმძლავრე
- მასშტაბურობა

- მარტივი პროგრამირება
- ინტელექტუალურობის შესასწავლად
- გეოგრაფიული გადანაწილება
- ფასის ეფექტურობა

ჭიანჭველების ამოცანაში აგენტთა ქცევები დაფუძნებულია მათ როლებზე. თვითოეული როლი განსაზღვრავს ჭიანჭველის ცოდნის ბაზის სპეციალიზაციას. მათი ცოდნის გადაცემა ერთმანეთის მიმართ მჭიდროდ არის დაკავშირებული ამოცანის ამოხსნის სისწრაფესთან. როგრც ვნახეთ, ამის მთავარი განმპირობებელია ცოდნის პირდაპირი გზით გადაცემა აგენტებისათვის.

რა მოხდება თუ ჭიანჭველები იმოქმედებენ გუნდურად და ექნებათ MAS? რას ნიშნავს ჭიანჭველების გუნდად გაერთიანება? რა განაპირობებს რამდენიმე ჭიანჭველის კოლექტივში შესვლას? გაცილებით უკეთეს შედეგს მივიღებთ, თუ მოვახდენთ ჭიანჭველათა გუნდურად ფორმირებას.

ჭიანჭველების ამოცანაში გუნდების ფორმირება ხდება დინამიურად. ნებისმიერ დროს, რეალურ დროში, რომელიმე კლასის ჭიანჭველამ შეიძლება დაიმატოს სხვა, იგივე კლასის ჭიანჭველა და მათი მოქმედება მოხდეს კოლექტიურად. ასევე, შეიძლება ვინმე გათავისუფლდეს გუნდიდან. თვითოეული კლასის ჭიანჭველათა გაერთიანებისთვის და გაცალკევებისათვის არსებობს გარკვეული პირობები, ასევე ამ პროცესის დადებითი და უარყოფითი მხარეები. გუნდში გაერთიანება უნდა მოხდეს ოპტიმალური პარამეტრის მხრივ, ოპტიმალური პარამეტრი ბევრი რამ შეიძლება იყოს, მაგალითად, მზვერავისთვის სწრაფად სიარული, ან თვალთახედვის რადიუსი, გადამზიდისათვის ჯამური ფოთლების აწევის ცონა, ან მოძრაობის სისწრაფე. განვიხილოთ აგენტთა ჯგუფურად გაერთიანების მაგალითები და კოლექტიური გადაწყვეტილებები.

ჯგუფური გადაწყვეტილებების მექანიზმი დაფუძნებულია ჯგუფურ არგუმენტებზე. არსებობს არგუმენტები, რომელთაგან ხდება "უფრო სწორის" არჩევა და მიიღება კოლექტიური გადაწყვეტილება. ბუნებრივია, ჩნდება კითხვა, რას ნიშნავს სწორი არგუმენტი? ვინ წყვეტავს რომელი არგუმენტი არის გამომავალი მონაცემი?

შავი დაფის არქიტექტურა არის რთული ამოცანების გადაჭრის სტრატეგია სხვადასხვა ცოდნის წყაროების გამოყენებით. შავი დაფა არის გლობალური წვდომადი მონაცემთა ბაზა რომელიც გამოიყენება შუალედურად, ამოცანის გადაჭრის ნაწილობრივ შედეგებზე. შავი დაფის არქიტექტურა შეგვიძლია წარმოვიდგინოთ როგორც ჯგუფი სპეციალისტების, რომლებიც სხედან შავი დაფის წინ. ისინი მუშაობენ ერთობლივად იმისთვის რათა მოახდინონ ამოცანის გადაჭრა და ამისთვის ისინი იყენებენ შავ დაფას როგორც სამუშაო ადგილს ამოცანის გადაწყვეტისთვის.

პრობლემის გადაჭრა იწყება მისივე გამოცხადებით და ხდება თავდაპირველი მონაცემების დაწერა შავ დაფაზე. სპეციალისტები უყურებენ შავ დაფას და ეძებენ შესაძლებლობას რომ წვლილი შეიტანონ ამოცანის განვითარებაში. როდესაც სპეციალისტი იპოვის შესაძლებლობას ის აკეთებს ჩანაწერს შავ დაფაზე და იმედოვნებს რომ ისინი გამოიყენებენ მის წვლილს რათა იპოვონ ამოცანის გადაჭრის გზა. ეს პროცესი გრძელდება მანამ სანამ არ მოხდება ამოცანის გადაწყვეტა. შავი დაფის არქიტექტურა შედგება სამი კომპონენტისგან.

შავი დაფის არქიტექტურა შედგება 3 კომპონენტისგან:

- ცოდნის წყაროები
- შავი დაფა
- მაკონტროლებელი კომპონენტი

ცოდნის წყაროები არიან დამოუკიდებელი მოდულები, რომლებიც შეიცავენ საჭირო ცოდნას ამოცანის გადაწყვეტისთვის. მათ არ სჭირდებათ იცოდნენ არსებობა სხვა ცოდნის წყაროების, მაგრამ მათ უნდა იცოდნენ, თუ რა მდგომარეობაშია ამოცანის გადაჭრის პროცესი და ინფორმაცია, რომელიც წარმოდგენილია შავ დაფაზე.

ჭიანჭველების ამოცანაში ნებისმიერ მზვერავს აქვს ფოთლის ამოცნობის R რადიუსი. მოცემული რადიუსის მქონე წრეწირის შიგნით იგი აღმოაჩენს ვარგის ფოთოლს და მონიშნავს მას. ყოველი მზვერავი მოძრაობს გარკვეული V სიჩქარით. დავუშვათ, ამოცანაში შემდეგი პირობაა - *ნებისმიერი მზვერავი ჭიანჭველისთვის რადიუსის დამოკიდებულება უკუპროპორციულია სიჩქარისა*. ჭიანჭველა რაც უფრო ჩქარა მოძრაობს, მით ნაკლებია მისი მოქმედების რადიუსი,

სოლო რაც უფრო ნელა მოძრაობს, პირიქით, მას შეუძლია უფრო ფართო ტერიტორია გამოიკვლიოს. მზვერავის მარგი ქმედების კოეფიციენტი გამოვთვალოთ შემდეგი ფორმულით:

$$Z = RV$$

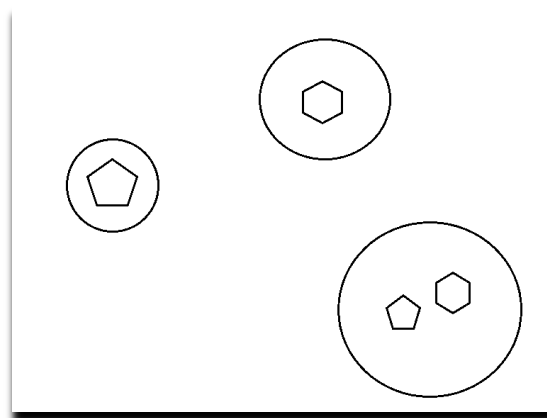
მზვერავების გუნდურად გაერთიანება მნიშვნელოვანია ამოცანის ამოხსნის წარმადობისთვის. თითოეული მზვერავის გუნდში აყვანის შემდეგ გუნდი იწყებს ახალი სიჩქარით მოძრაობას. ყველა ჭიანჭველას საშუალოსიჩქარეა

$$V_{max} = \sum_{i=1}^n (V_i / n)$$

სოლო ჯამური დასაზვერი ფართობის რადიუსი, თითოეული ჭიანჭველის მიერ გამოკვლეული ფართობის რადიუსების ჯამის საშუალო არითმეტიკულია:

$$R_{max} = \sum_{i=1}^n (R_i / n)$$

ერთის მხრივ, მზვერავებმა უნდა გაითვალისწინონ ამოცანის შესრულების სისწრაფე და გუნდში ჰყავდეს სწრაფი გადაადგილების მქონე ჭიანჭველები. მეორეს მხრივ, ფართე რადიუსი, რადგან უფრო მეტი ხილვადობა ჰქონდეს ჯგუფს. ამ შემთხვევაში, საჭიროა ოპტიმალური შუალედის პოვნა, ჭიანჭველები არც ძალიან უნდა აჩქარდეს და არც უნდა შენელდეს. შესაბამისად, მათ უნდა ჰქონდეს მაქსიმალურად დიდი ხილვადობის რადიუსი და უნდა მოძრაობდეს მაქსიმალურად სწრაფად (სურ 6).



სურ. 6 მზვერავთა გუნდური ფორმირება და საერთო რადიუსი

თვითოეულ აგენტთა ჯგუფზე შეგვიძლია განვიხილოთ კლასიკური შავი დაფის არქიტექტურა. აგენტთა არგუმენტები იწერება შავ დაფაზე

და მაკონტროლებელი მექანიზმი წყვეტს საბოლოო გამომავალ გადაწყვეტილებას.

ყოველი ახალი ჭიანჭველას შემხვედრის შემთხვევაში, ირთება გუნდის არგუმენტაციის მექანიზმი, თვითოეული ჭიანჭველა ამბობს სათავესო არგუმენტს მისი მარგი ქმედების კოეფიციენტიდან გამომდინარე, ანუ რომელი პარამეტრით არის უპირატესი. შავ დაფაზე დაწერილი არგუმენტების მიხედვით, მაკონტროლებელი მექანიზმი წყვეტს გუნდიდან წამოსული არგუმენტების მიხედვით აიყვანოს თუ არა ახალი წევრი, თუ ჩაანაცვლოს არსებული. ახალი წევრის აყვანამ დიდი რადიუსით, შეიძლება ძალიან შეანელოს მოქმედება, ხოლო სწრაფად მოძრავმა ჭიანჭველამ კი ხილვადობის რადიუსი შეამციროს.

აღწეროთ გაძლიერებითი დასწავლის ალგორითმი, სადაც თვითოეულ ჯერზე გადაწყვეტილების მიღება დაფუძნებულია ჯილდოზე. ჯილდო პირობითად, არის მიზნის მიღწევის შედეგად მიღებული სარგებელი. ჩვენს კონკრეტულ შემთხვევაშიც გურუები, ზუსტად ასეთი მეთოდით მოქმედებს.

თუ ჩვენ გვაქვს არსებული მდგომარეობა x და ხელმისაწვდომი მოქმედება a , მაშინ Q დასწავლითი ალგორითმი მოიცემა ბოლცმანისტოლობით:

$$p(a_i|x) = \frac{e^{Q(x,a_i)/T}}{\sum_{k \in \text{actions}} e^{Q(x,a_k)/T}}$$

სადაც T არის ტემპერატურის პარამეტრი და ის განსაზღვრავს გადაწყვეტილებათა მიღების შემთხვევითობას. გურუ იღებს გადაწყვეტილებას, იღებს r - ჯილდოს. შემდეგ კი გადადის ახალ მდგომარეობაზე. ყოველ ჯერზე აგენტი ანახლებს $Q(x, a)$ - რეკურსიულად, უმატებს პოზიტიურობის წონას β და აკლებს სამომავლო რესურსებს:

$$Q(x, a) \leftarrow Q(x, a) + \beta(r + \gamma V(y) - Q(x, a))$$

γ ($0 \leq \gamma \leq 1$) არის კლებადობის პარამეტრი და $V(x)$ მიიღება ასე:

$$V(x) = \max_{b \in \text{actions}} Q(x, b)$$

გაძლიერებითი დასწავლა თვითოეული აგენტისთვის ამომავალი არგუმენტია, რათა უპირატესი იყოს მისი აზრი შავ დაფაზე. როგორც

ვთქვით, შავი დაფის მაკონტროლებელი მექანიზმისთვის ყველაზე მთავარი ფუნქციაა ოპტიმალური არგუმენტის პოვნა. კონკრეტულ მაგალითში საუკეთესო მეთოდი იქნებოდა ხარბი ალგორითმების გამოყენება.

ხარბი ალგორითმების (greedy algorithms) გამოყენებისას, ყოველ ბიჯზე კეთდება ლოკალურად ოპტიმალური არჩევანი იმ იმედით, რომ საბოლოო შედეგიც ოპტიმალური იქნება. ეს მიდგომა ყოველთვის არ ამართლებს, მაგრამ ზოგიერთი ამოცანისათვის მართლაც ძალიან ეფექტურია. რაც მთავარია, არსებობს სტანდარტული პროცედურები იმის გასარკვევად, კორექტულად იმუშავებს თუ არა მოცემული ამოცანისთვის ხარბი ალგორითმი.

ჭიანჭველების ამოცანაში ხარბი ალგორითმის შემავალი პარამეტრი ორნაირი შეიძლება იყოს, ეს პარამეტრები განსაზღვრავს გამომავალ ოპტიმალურ ვარიანტს. მაგალითად, მზვერავის შემთხვევაში შეიძლება იყოს რადიუსი, ან სიჩქარე. ჩვენ წინასწარ უნდა განვსაზღვროთ უპირატესი პარამეტრი, გვინდა, რომ ჭიანჭველამ სწრაფად იაროს, თუმცა მინიმალური რადიუსი ჰქონდეს? თუ პირიქით, იაროს ნელა და დიდი ხედვის რადიუსით.

ყოველი შემომავალი მონაცემი უნდა შედარდეს არსებულებს და იქიდან აირჩევა მაქსიმალური, განვიხილოთ ფუნქცია *GREEDY_ACTIVITY_SELECTOR(s,f)* ყოველი შემომავალი ორი მასივისთვის:

```
n = length[s];
A = {1};
j = 1;
for(i = 2;i<=n;i++)
    if(si>=fi)
        A+={i}
    j = 1;
return A;
```

მზვერავთა შემთხვევაში, შავი დაფისთვის ასარჩევია გადაწყვეტილება, როდესაც გუნდში ხდება ახალი მზვერავის აყვანა,

აიყვანოს თუ არა ჭიანჭველა გარკვეული რადიუსით, თუ გარკვეული სისწრაფით. ამ ფსევდოკოდის მიხედვით მოცემული კონკრეტული პარამეტრი, სიჩქარე, ან რადიუსი იქნება ოპტიმალური არჩევანი.

მზვერავთა ჯგუფური გადაწყვეტილებების მსგავსად შეგვიძლია განვიხილოთ ფოთლის მჭრელთა დინამიკური დაჯგუფებებიც. მათი კოლექტივიზაციაც დამოკიდებულია მარგი ქმედების კოეფიციენტზე, რომელიც გამომდინარეობს მათსავე დამახასიათებელი ქცევიდან - შესაბამისი სიმბოლოს ფოთლის დანაწევრებისგან.

ჭიანჭველათა ჯგუფი, რომელთა შემადგენლობაში შედის a, v, g, n, \dots ფოთლის ამომცნობი ჭიანჭველები. თითოეული ჭიანჭველათა ჯგუფისათვის არსებობს ჯგუფური მარგი ქმედების კოეფიციენტი β კეშ-ცხრილი(ცხრ.1), სადაც წერია შესაბამისი სიმბოლოს ამომცნობი ჭიანჭველის არაწარმადობის ან წარმადობის კოეფიციენტი. თავიდან ეს კოეფიციენტები არის 0-ის ტოლი.

ყოველი ახალი სიმბოლოს ამომცნობი ჭიანჭველის დამატების შემთხვევაში ამ კეშ-ცხრილს შეემატება შესაბამისი სიმბოლოს ერთი ჩანაწერი, მნიშვნელობით 0. განვიხილოთ შემთხვევები თუ როგორ იცვლება კეშ-ცხრილი სიმულაციის ყოველ შემდეგ ნაბიჯზე.

a	0
v	0
g	0
n	0
...	0

ცხრილი1 ფოთლის მჭრელთა საწყისი პარამეტრების ცხრილი

ნებისმიერი ფოთლის დამუშავების შემდეგ, ჯგუფში გადაიხედება შესრულებული სამუშაო და ამ სამუშაოზე მომუშავე ჭიანჭველები გადაირჩევა. ვთქვათ, თუ ჭიანჭველა a -სთვის შესაბამისი სიმბოლო ვერ მოიძებნა და ის, ასე ვთქვათ, უსაქმოდ დარჩა, კეშ-ცხრილის გრაფაში მისი სიმბოლოს შესაბამის მნიშვნელობა შემცირდება 1-ით. ესე იგი, არსებული ჭიანჭველის წარმადობა შემცირდა. თუ დარჩა ისეთი ფოთლის ნაწილი, რომლის სიმბოლოს ამომცნობი არ მოიძებნება გუნდში, მაშინ გუნდის კეშ-ცხრილში ჯერ დაიძებნება შესაბამისი

სიმბოლოს ჩანაწერი. პონის შემთხვევაში მისი მნიშვნელობა გაიზრდება 1-ით, ხოლო არ-პონის შემთხვევაში, გაკეთდება ახალი ჩანაწერი ამ სიმბოლოთი და მნიშვნელობით 0. ეს იმას ნიშნავს, რომ შესაბამისი სიმბოლოს ამომცნობი ჭიანჭველის საჭიროების კოეფიციენტი გაიზარდა, ან საჭირო ჭიანჭველის გრაფა დაემატა. (ცხრილი 2)

<i>a</i>	1
<i>v</i>	0
<i>g</i>	2
<i>n</i>	-3
...	0
<i>t</i>	0

ცხრილი2 ფოთლის მჭრელის ჰეშ-ცხრილის გარკვეული ცვლილებების რომელიმე ბიჯზე

პირობითად, არსებობს გარკვეული ზღვარი ჰეშ-ცხრილის მნიშვნელობებში, როდესაც საჭირო ხდება გურუ-ჭიანჭველის მიერ ახალი წევრის დამატება ან ძველი წევრის გაშვება. თუ მარგი ქმედების კოეფიციენტი მნიშვნელობით გარკვეულ ზღვარს (- δ) ჩამოსცდება (ცხრ.2), გურუ იღებს გადაწყვეტილებას, რომ შესაბამისი სიმბოლოს ამომცნობი ჭიანჭველა საჭირო აღარ არის ამ გუნდში, და იგი მას ათავისუფლებს. ამ ჭიანჭველის შესაბამისი სიმბოლოიანი ჩანაწერიც იშლება ჰეშ-ცხრილიდან. თუმცა გათავისუფლებული ჭიანჭველა შეიძლება საჭირო იყოს სხვა გუნდისთვის, რომლის გურუმაც მისი სიმბოლოს მქონე ჭიანჭველის დამატების გადაწყვეტილება მიიღო, რაც შემდგენაირად განისაზღვრება - თუ სიმულაციის გარკვეული ბიჯების შემდეგ რომელიმე სიმბოლოს კოეფიციენტი გასცდა გარკვეულ ზღვარს(δ), ე.ი. საჭიროა ამ სიმბოლოს ამომცნობი ჭიანჭველის დამატება გუნდში, რადგან მისი სამუშაო უხვად არის შესაბამის კოლონაში.

- ყოველი შემდეგი ჭიანჭველის დამატების ფსევდო კოდი

foreach(each i; i < HashTable.length; i ++)

$\{if(i.value > \delta) AddNextAntInColony(i.key)\}$

- არასაჭირო ჭიანჭველის გათავისუფლება კოლონიიდან

$foreach(each\ i; i < HashTable.length; i++)$
 $\{if(i.value < -\delta) RemoveAntFromColony(i.key)\}$
 $\{DecrementHashTable(i.key)\}$

დისერტაციის ზოგადი დასკვნა

დისერტაციაში განვიხილეთ მრავალაგენტიან ვიდეოთამაშებში აგენტთა კოლექტიური გადაწყვეტილებები. მიმოვიხილეთ თამაშების უანრები. ვნახეთ, რომ უანრების მიხედვით ხელოვნური ინტელექტის ამოცანები ერთმანეთისგან განსხვავდება. თუმცა პრინციპი ერთია, აგენტის მოქმედება დინამიურ ან სტატიკურ გარემოში. აგენტთან ერთად კი, რაც მთავარია, მოთამაშის მოქმედება. მოთამაშე მთავარი ასპექტია ვიდეოთამაშში, ვის გარშემოც ტრიალებს მთელი ხელოვნური ინტელექტუალური სისტემა და ინტერაქცია.

განვიხილეთ აგენტთა სისტემები, ერთაგენტიანი, მულტიაგენტური. ასევე ინტელექტუალურ აგენტთა თეორია, აღვწერეთ რას ნიშნავს ერთაგენტიანი და მულტიაგენტური სისტემა, რითი განსხვავდება ერთმანეთისგან, რა მსგავსებებია და რა საჭიროების შემთხვევაში გამოიყენება ერთი, ან მეორე.

კოლექტიურ გადაწყვეტილებების გარჩევამ დაგვანახა ცნობილი პრობლემა ვიდეოთამაშებში და ზოგადად, ხელოვნურ ინტელექტში, ცოდნის გადაცემის პრობლემა. ვნახეთ რა მდგომარეობაა დღესდღეობით თანამედროვე ვიდეოთამაშებში ამ მხრივ, წარმოვადგინეთ ჩვენი ვერსია ცოდნის გადაცემისა, ცოდნის მარკირების ამოცანა. ვნახეთ, რომ აგენტთა როლების გადანაწილებით და ერთდროული მოქმედებით გაცილებით სწრაფად და ადვილად იხსნება რთული სისტემები. ეს ნათლად დაგვანახა ჭიანჭველების ამოცანამ, რომელსაც თამაშის სახე მივეცით.

ჭიანჭველების ამოცანაში შემოვიტანეთ სამი კლასი, ამ სამი კლასის დაფუძნებით მოცემული ამოცანა დაიყო ქვეამოცანებათ, ქვეამოცანების

შესრულება დამოკიდებული გავხადეთ თვითოეული კლასის ჭიანჭველაზე, რომლებიც ერთმანეთს ცოდნის მარკირებით ეკონტაქტებიან.

თემაში განვიხილეთ ასევე ვიდეოთამაშების არგუმენტაციის მექანიზმი, რომელიც მთავარი ამოსავალი წერტილია კოლექტიური გადაწყვეტილებებისა. ჩვენი არგუმენტაციის მექანიზმის იმპლემენტაცია ჭიანჭველების ამოცანაშიც წარმოვადგინეთ და ვაჩვენებთ, თუ როგორ შეიძლება მოახდინოს გარკვეული კლასის ჭიანჭველამ გუნდის ფორმირება, რაც მთავარია, რეალურ დროში მიმდინარეობს.

ჩვენს ნაშრომში აღწერილი თემა გამოსადეგია ნებისმიერი მსგავსი ამოცანისთვის, სადაც საჭიროა მულტიაგენტური ტექნოლოგია. ვიდეოთამაშების სხვა დეველოპერებს შეუძლიათ პირდაპირ გამოიყენონ ეს ნაშრომი, რათა განახორციელონ ცოდნის მარკირების ამოცანა. ასევე შეუძლიათ აგენტთა გუნდის ფორმირება ჯილოდოს პრინციპზე დაფუძნებით.

ეს ამოცანა გამოდგება არამარტო ვიდეოთამაშებში არამედ სხვა სფეროშიც. საჭიროა განვახორციელოთ ამ ამოცანის უნიფიცირება, როგორც უნიკალური ფრეიმვორქი, რომელიც ერგება ნებისმიერი სფეროს, სადაც მრავალაგენტიანი, მულტიპროცესულ, ხელოვნურ აგენტთა მოქმედებაა საჭირო.

დისერტაციის თემასთან გამოქვეყნებული

პუბლიკაციები

1. გიორგი აბელაშვილი, ზურაბ ბოსიკაშვილი, "ვიდეოთამაშებში მულტიაგენტური დასწავლის მეთოდების შესახებ" - მართვის ავტომატიზებული სისტემები, N2 (20), თბილისი, 2015 წელი, გვერდი 52. სტატიის ლინკი: [http://www.gtu.ge/Journals/mas/Referat/ASU-2015\(2_20\)/52_57.pdf](http://www.gtu.ge/Journals/mas/Referat/ASU-2015(2_20)/52_57.pdf)

2. გიორგი აბელაშვილი, ზურაბ ბოსიკაშვილი, "ხელოვნური ინტელექტის არგუმენტაციის მექანიზმის გამოყენება ვიდეოთამაშებში" - მართვის ავტომატიზებული სისტემები, N2 (20), თბილისი, 2015 წელი, გვერდი 58. სტატიის ლინკი:

[http://http://www.gtu.ge/Journals/mas/Referat/ASU-2015\(2_20\)/58_65.pdf](http://http://www.gtu.ge/Journals/mas/Referat/ASU-2015(2_20)/58_65.pdf)

3. გიორგი აბელაშვილი, "ვიდეოთამაშებში აგენტთა გაერთიანებისა და კოლექტიური გადაწყვეტილებების შესახებ", მართვის ავტომატიზირებული სისტემები, N2 (21), თბილისი, 2016 წელი, გვერდი 138.

Abstract

Computer and information technology for the development of the modern stage set up videogame industry. A People play with computers (agents) in a virtual environment. Playing is very interesting when agents acts more intellectually. But more interesting, when computer Ai - agents join together and they collectively play with people. Collective decision-making is one of the important tasks of modern artificial intelligence field and also in videogames

There are lots of video game genres. Videogames includes many interesting and mesmerizing AI-problems. These problems, most often are based on Individual ai-agents theory. Individual agents operate independently, they are only for the specific environment, there are a variety of such agents, they do not communicate, do not share, or receive knowledge from, or any information about each other.

There are specific cases when the problem can be solved more quickly and optimally with multiagent ai-system, i.e. the union of the individual agents.

When agents are in union, many of their decisions aren't individual, but also collective. Collective decision is, when a main knowledge based on each individual knowledge of each agents of some group. Agents in group are not always individual, because they communicate each other, they exchange information and

knowledge each other. The goal of this article is to create optimal decision-making solution for collective system of agents. The purpose of this system will show that some of the approaches to the solution of multiagent-Ai-system is more optimal and dissolves rather than individual agents in the individual ai-system. The paper used in our study can be used by other games, or similar artificial intelligence problems, where multiagent system approach is more effective.

So, Artificial Intelligence is one of the important issues discussed concerning collective decisions, reasoning mechanism. Argumentation mechanism takes huge place in modern Ai. Agents say some arguments based their own knowledge. There is a special argument-controlling-mechanism which controls collective decisions.

The starting point of argumentation mechanism is knowledge. Artificial intelligence agents gather some knowledge of the Games, based on the facts and circumstances. This knowledge is stored in the knowledge base. Individual agent's case, knowledge is collected individually - agent learns only with its individual acting. It doesn't communicate with other agents, it doesn't exchange any information or knowledge with other agents. But agents union collects common knowledge. They communicate each other with special communication mechanism. Unlike individual agents, they transfer each other's knowledge and some information. The collective knowledge of the agents group model and they exchange of information needed for a flexible mechanism for the transfer of knowledge.

Nowadays, in videogames there are many ways to transfer knowledge. We consider one version - Problem of Marking of Knowledge.

In this article we develop argumentation mechanism based on Problem of Marking of Knowledge, and this one also provides some collective decisions for our article. In Problem of Marking of Knowledge, agents leave some form of knowledge to each other on certain positions of game world. And the other agents get this type of knowledge directly in their database of knowledge. They do not have to use special algorithms to calculate some of the facts and conditions to gain

this kind of knowledge, because this knowledge is already final, final to "remember". We also discuss the advantages and disadvantages of this method.

In this paper we have the example of the ant task. The task is considered as a specific video game, which is essential for optimal results. There is a two-dimensional plane. On the plane appear certain types of objects, including the leaves. The leaves contain a simple alphabetical letters. On the plane there are also the ant-agents, whose goal is to collect the leaves and take to repository. Each ant is able to recognize one of the letters on the leaf, if it found it, it cut the letter-character and takes in repository. The game goal is to take leaves as much as possible in special given time.

The task we have as individual agents in the system, as well as - in multiagent. Individual agents will take on the classic individual theory of ant activity - they should check every object to find leaves. After finding the leaf object, they must check symbol contain test. After this they cut the symbol and they take it in repository. And with multiagent system we will try to solve problem with group of ants.