Multi-Agent Decision-Making Model for the Construction of Schools

Kele Mbbe Ripaul Carlos
Faculty of Science, University of Ngaoundéré, Cameroon
PO BOX 13750
PO BOX 454

Nkenlifack Marcellin Julius
Faculty of Science, University of Dschang, Cameroon
PO BOX 067
PO BOX 96

Kamla Vivien Corneille
Dept of Mathematics and Computer Science
ENSAI, University of Ngaoundéré, Cameroon
PO BOX 454

ABSTRACT
The model presented in this article will help to make a decision for the construction of a school (school / high school / university). It is based on agents; each playing a role in the construction process of the establishment. The context of the problem of the construction of a school lies in the finding of a saturation of schools of a locality to the point where the numbers of learners in classrooms become plethoric and their follow-up by teachers becomes difficult causing a rate of high failure and low educational level of learners. These one no longer have possibility to enroll. The model must indicate to the authority in charge of management of educational system the areas where it is necessary to build schools. For this reason it takes into account the maximal distance that a learner must cover by foot from his home to the school. It takes also into account the number of learners by educational level who haven’t possibility to enroll because the number of learners in classrooms is plethoric. The authority in charge of the management of education must then evaluate material, financial and human resources allowed at its disposal in order to decide whether to build a school or not in this locality.

Agents have been defined for schools, learners and decisions makers to describe the behavior of schools, learners and education authorities respectively.

General Terms
School, decision-maker, model, algorithm, learner, teacher, system.

Keywords
Multi-agents model, complex system, agent, multi-agent system, school construction.

1. INTRODUCTION
Making decision today can be a development factor when this one is done properly; if not it can be the cause of a failure for a project or for the development of a society. It intervenes in all areas of daily life namely family, aeronautic, health, agriculture, management, education, etc.…, No sector is spared. A business manager, a credit manager, a minister or a head of state needs to make decision for the proper functioning of the structure or country he leads. For this purpose he needs a making decision tool. However some of these tools are based on the modeling of complex system. All the new models take environmental, economic, social and anthropic aspects into account. They help to better understand the functioning of the world.

Education is a sensitive sector in societies and those in charge of it should make rational decision in order to help theirs populations to better educate. Building a school is not easy. It is important to take ownership of a making decision tool but many researchers haven’t looked much into this area. For this reason it is the duty in this article to highlight all elements needed to design a decision making aid model for the building of a school

The following section presents standards to respect for the construction of schools, the second and third section present respectively decision support and the multi-agents systems. The fourth, a multi-agents model which will help to make a decision for the construction of schools in a locality followed by a multi-agents simulation based on the simulation platform GAMA. Finally the conclusion and perspectives will be presented.

2. STANDARDS TO RESPECT FOR THE CONSTRUCTION OF SCHOOLS

2.1 Number of learners in classrooms
The maximum enrollment is 40 or 42 students per classroom according to the classroom model. The standard provides classes with three frontal tables-benches of 42 students and classrooms with four frontal tables-benches of 40 students [1]. So, a ratio of 40 students per teacher and 25 students for preschool [3].

2.2 Location of the school
In rural areas, the location chosen should be easily accessible for all learners attending the school. They should not cross or run along national roads or dangerous rivers. It will be avoided to choose places close to any other particular situations considered (locally) as constituting a physical risk [2].

In cities, dust, gas, odors and noise must be avoid to bother learners. To this effect, schools should not be located too close to factories, airports, petrol stations or arteries.

School grounds must be located at least 400 m away from nuisances: odors (slaughterhouse, henhouse, pigsty, tanneries, sugar refineries, distilleries, gas station ...), smoke (factories), dust (cement plants, mills, arteries too much frequented), and noise (market, factories, public squares, busy road, churches, and airport)[2].

The minimum distance to be respected between schools and unhygienic environments (wild or public waste, hospital, cemetery, garage, ponds, etc.) is about 100 m [2].

School locations should be outside risk areas such as [2]:

- sites too exposed to winds
- flood zones:
- areas prone to landslides, falling rocks, mudslides due to the saturation of the ground in the event of a cyclone, cliff collapse or rock fall in the mountainous regions in

the event of an earthquake

- areas exposed to risks, in general:
- under an electrical line that can break in the event of a cyclone and cause risks of electrocution and fire,
- in the vicinity of the sea within 800 m of the coastline,
- near tall trees,
- on landfills: risk of explosion, fire or contamination,
- on certain types of soils[2]
- sites presenting risks of liquefaction,
- unstable ground,
- presence of alluvium of variable thickness at the foot of the slope or significant thickness in the middle of valleys (likely to amplification).
- presence of different geological formations: at the boundary between rocky and soft soils (the soil will settle unevenly, it is called differential settlement)
- heterogeneous soil made of pockets of compressible clay embedded in a gravelly soil.

The minimum land area prescribed for the construction of a school must be large enough to contain the school infrastructure and allow for future extension of the building, ie at least its double. The total area of a school site, excluding sports facilities, is to be calculated on the minimum basis of [2]:

- 25 m² / student in rural areas.
- 12.5 m² / learner in an urban environment (the construction in floor reduces the total area required.)
- 15 to 20 m² per learner depending on the shape and model of the ground including play facilities, for preschool.

3. DECISION SUPPORT

The decision is perceived as the exercise of a choice between several possibilities of actions at a given moment [3]. Those who make this choice are "decision makers". The decision-maker thus becomes responsible for his choice and must be able to assume the consequences of his choice. As Roy [4] defines it, "Helping to decide is first and foremost helping to clarify the formulation, the transformation and the argumentation of the preferences." At this level, the key concept is that of the criterion ". This single-criteria approach assumes that the system is simple enough that it is possible to evaluate the different possibilities of action using a single indicator. The choice to consider only one criterion amounts to explicitly reducing the reality to one dimension. In the case of complex systems where several dimensions must be taken into account, the decision aid changes to become according to Roy [4] "the activity of the one who, based on models clearly explained but not necessarily completely formalized , helps to obtain elements of answer to the questions an actor poses in a decision-making process, elements that help to take the decision and normally to recommend, or simply to favor, behavior likely to increase the coherence between the evolution of the process on the one hand, the objectives and the value system on which these actors are placed on the other." The implemented models are often based on mathematical concepts, diagrams, languages, agents etc... Multi-agent systems are intrinsically knowledge-based systems.

4. MULTI-AGENTS SYSTEMS

Multi-agent systems are intrinsically knowledge-based systems. As knowledge representation systems, they can model both how to encode knowledge and how to use it. The construction of a decision support using multi-agents oriented models involves the mobilization of knowledge that is spread over several levels. The different levels are successively mobilized during the process of finding solutions and helping decision-making: from the analysis of the problem as a whole, to the mental states of the agents [3].

4.1 Definition of an agent

An agent is defined as a physical or virtual entity [5]:

- who is able to act in an environment,
- who can communicate directly with other agents,
- which is driven by a set of trends (in the form of individual objectives or a function of satisfaction, or even survival, which it seeks to optimize),
- who has own resources,
- who is able to perceive (but in a limited way) his environment,
- which only has a partial representation of this environment (and possibly none),
- who has skills and offers services,
- which can possibly reproduce,
- whose behavior tends to meet its objectives, taking into account the resources and skills at its disposal, and according to its perception, representations and the communications it receives.

From this definition it follows that a physical entity is something that is capable of acting in the real world; that is to say, capable of producing an effect in its environment Example: a drug, a school are physical entities; on the other hand, a software or a software component are virtual entities, because they do not exist physically. An agent is therefore active; he has the possibility of answering in the affirmative or refusal to requests from other agents [5]. To act, he needs a number of resources.

4.2 Definition of an MSA

A multi-agent system (or MSA) is a system composed of the following elements:

- An environment E, that is to say a space generally having a metric.
- A set of objects O. These objects are located, that is to say that, for any object, it is possible, at a given moment, to associate a position in E. These objects are passive, i.e. they can be perceived, created, destroyed and modified by agents.
- A set A of agents, which are particular objects ($A \subseteq O$), which represent the active entities of the system.
- A set of relations R which unite objects (and therefore agents) between them.
- A set of Op operations allowing A's agents to perceive, produce, consume, transform and manipulate O objects.
- Operators responsible for representing the application of these operations and the reaction of the world to this

attempt to modify, which will be called the laws of the universe [5].

The environment can be [6]:

- accessible or inaccessible: an environment is said to be accessible when an agent can obtain all the informations on the environment or more simply all of those that are necessary for him to decide and act.
- Deterministic or stochastic: in a deterministic environment there is no place for random phenomena, an action will always have the same consequences.
- Static or dynamic: an environment is said to be static when it does not have its own dynamic, in this case only the agents can modify it.
- Discreet or continuous: an environment is discreet when there is only a finite and fixed number of possible perceptions and actions on it.

Ferbert has added two other properties [5] to that cited above:

- Centralized: An environment is centralized when all agents have access to the same data structure containing all environmental information.
- Distributed: An environment is distributed when it is composed of a set of cells arranged in a network. Each cell behaves like a mini centralized environment.

4.3 Classification of agents

The agents depend on the studied system, these can be virtual or physical, located or not in their environment, mobile or static, communicating or not [3]. The classification of agents could include a multitude of categories. The research carried out has led to classify the agents according to the nature, their behavior. There are today only three main categories [3]:

- The reactive agents.
- Cognitive agents.
- Hybrid agents.

4.3.1 Reactive agents

Reactive agents act according to their perception of changes in their environment, the messages they receive, their mental state and their goals. They are characterized by a lack of representation of their environment. They react simply reflexively according to external stimuli and their immediate needs [3].

The reactive agents are, because of the simplicity of their behavior, the easiest to conceive [3]. They are used to model entities with simple behaviors such as sea urchins, ants, air conditioning or firewall.

4.3.2 Cognitive agents

Cognitive agents have important reasoning skills. Their behavior is not limited to responding immediately to stimuli coming from outside. They have a representation of their environment and accumulate experience. To achieve their goals, these agents are able to build plans and cooperate. They use the experience they have gained to make decisions. In many systems[3]. Cognitive agents can represent entities as diverse as a supermarket customer, a trader, a lobby or an IP router.

4.3.3 Hybrid agents

Hybrid agents are a compromise between pure cognitive agents that imperturbably follow the plan they have established, and strictly reactive agents that only make decisions based on data from their sensors. Hybrid agents have a partial representation of their environment, they know how to develop and follow a plan to meet their desires [3].

Hybrid agents are those who possess the behavior closest to the natural behavior of man and the higher animals who are able to abandon for a time their main task to take advantage of an opportunity [3].

4.4 Agent and object comparison

At first sight, it is not easy to distinguish between an agent and a computer object. Indeed, they all have attributes and methods to handle them.

Wooldridge [7] identifies three main differences between object and multi-agent approaches:

- Unlike objects, agents are autonomous. Thus, when an object receives a message, a method is executed, whereas in the case of an agent the receipt of a message does not systematically lead to the execution of an action. The agent is free from his actions and reacts only if an action brings him closer to his goals.
- An agent is characterized by his flexible behavior (reactive, proactive: he acts to achieve his goals, social: he interacts with other agents) and autonomous. This type of characteristic is not necessary for the definition of an object. Although it is possible to develop object-oriented programs that integrate this notion of autonomous behavior, this notion is foreign to the standard of object programming.
- Agents generally have their own Thread, they all act at the same time: they carry out their activities in parallel. Objects typically share a single Thread, the execution of an object-oriented program is basically sequential.

4.5 Interactions between agents

Interactions are reciprocal actions that modify the behavior or the nature of the elements, bodies, objects, phenomena in the presence or influence. In a multi-agent system, interactions are defined as the dynamic linking of two or more agents through reciprocal actions. The different situations of interactions between agents according to Jacques Ferber are classified as follow [5]:

- Independence: The situation of independence poses no problem from the point of view of multi-agent and comes down to the simple juxtaposition of the actions of agents taken independently, without there being any interaction. For example, people who cross in the street knowing that there is enough place to cross. This is a neutral situation that does not require any specific interaction.
- Simple collaboration: Simple collaboration is a simple addition of skills that do not require additional coordination actions between stakeholders.
- Overcrowding: overcrowding is characteristic of all situations in which agents interfere with each other in performing their tasks when they do not need each other. An example is that of car traffic,
- Coordinated Collaboration: Complex collaboration requires that agents coordinate their actions in order to have the synergy of all of their skills. As an example we can cite industrial activities requiring a distributed approach, such as network control, design and manufacturing of industrial products.

- Pure Individual Competition: When goals are incompatible, agents must fight or negotiate to achieve their goals. The "sport" competition supposes that the agents have all the same resources at their disposal and that they are placed in identical initial situations. Simply, let the best win.
- Pure Collective Competition: When agents do not have sufficient expertise, they must come together in coalitions or associations to achieve their goals. This grouping is carried out along a double movement: the first tends to bind the individuals within groups united by links of coordinated collaboration and the second to oppose the groups between them. A typical example of this type of situation is given by team competition, such as the team relay race.
- Individual Conflict for Resources: When resources cannot be shared, there is a typical conflict situation whose resources are at stake, each wanting to acquire them for himself alone.
- Collective Conflict for Resources: This type of situation combines collective competition with individual conflicts for resources. Coalitions struggle against one another to gain the monopoly of a good thing, a territory or a position.

5. PRESENTATION OF A MULTI-AGENTS MODEL FOR THE CONSTRUCTION OF A SCHOOL

5.1 The role of the model

The model will allow the decision-maker in charge of national education the areas where it is necessary to build a school.

5.2 Characteristics of the model

Figure 1 shows the different agents of the model; they are:

- learner,
- person in charge of building schools
- Establishment.

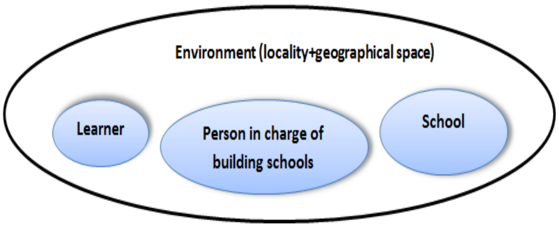

Fig 1: Agents of System

This allows presenting the architecture of the model. It is described in the figure 2 as follows:

1. The quest for registration: learners are looking for institutions to register,

2. Number of learners control: school through his responsible check each learner's application for enrollment if the numbers provided for by the regulations in force are respected or not;

3.a learner acceptance: learner registration is favorable because there are still places and they can access the school;

3.b. refusal of learners: no places

4. Indication of areas where to build schools: recognized areas for building a school

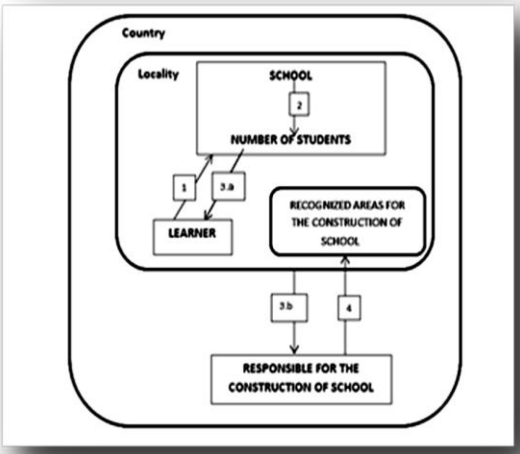

Fig 2: The system architecture

From this architecture the functions performed by each agent within the model in the institutions building process are noted as follow:

- The learner can: search for an establishment, register.
- The establishment may: increase, decrease, be overcrowded, and accept a learner.
- The person in charge of the construction of the establishments can: choose a place of construction of an establishment, build an establishment.

5.3 Implementation of each function

5.3.1 The function "look for a school"

The search for a school is an operation that every learner is called upon to perform. The latter should look for a school in the following cases:

- when there is a new resident in the locality: we take for example the case of students looking for a university after the baccalaureate; students who have obtained the transfer from one locality school of departure to another of an arrival city,
- when he has been dismissed from a school in the city where he resides,

The algorithm is in the following form:

Global: NB_School the number of schools in the locality;

Type TAB = Array [1..NB_Schools] of schools

Local: **i** variable counting schools of the locality,

Find a Boolean variable that indicates whether the learner has found a school.

T variable of type TAB

Dmax Maximum distance a learner must travel

A variable representing the learner

Function: place Function to check if there is class for registration.

1. **Begin**

2. i = 1

3. Find = false

4. While (i <= NB_School) AND (dist (T [i], A) <Dmax) do

5. if (No (place (T [i], level)) then

6. i = i + 1

7. else

8. Find = True

9. endif

10. endwhile

11. return (Find)

12. **End**

5.3.2 The "enrollment" function

In order to register, the learner must be a former learner from his school or have been admitted as a new learner at this school. The algorithm is as follows:

Global: **L** the level of learning of the learner;

N the number of learners who do not have a school to enroll in Level L

A learner who wants to write

E school where the learner s

Ins Boolean to tell if the learner is registered or not

Function: **Search** function helping learner to search for a school.

1. **Begin**

2. Ins = false

3. if (search (A, E, L) == True) then

4. N = N - 1

5. Ins = true

6. endif

7. return Ins

8. **End**

5.3.3 The function "signal a congestion"

The report of a congestion is made by the head of school in the case where the maximum of the number of leaners is reached and exceeds the threshold fixed by the standard. The algorithm is as follows:

Global: **Nmax** the maximum number of learners per level of study

M the number of classes

I the counter of study levels

Type **TAB2** = Array [1..M] of Class

Local: T2 variable type TAB2

1. **Begin**

2. For I = 1 to M do

3. If number (T2 [I])> Nmax then

4. Show ('alert on' I)

5. break

6. endIf

7. EndFor

8. **End**

5.3.4 The function "accept a learner"

In the case where there is still classroom for a level requested by a learner then if the latter fulfills the enrollment requirements, it is accepted by the school head. The algorithm is as follows:

Local: a the learner

E institution requested by the learner

r variable which allows to know the result of the research of a on a level of E

A Boolean variable indicating the acceptance of a

L level of study requested by the learner

Function: search function that indicates whether or not the learner found a school

condition that checks if the learner fulfills a number of conditions

Begin

1. A = False

2. r = search (a, E, L)

3. if (r == True) then

4. if (condition (a)) then

5.A = True

6. endif

7. endif

8. return (A)

9. **End**

5.3.5 The "increase" function

Whenever there is a new learner enrolled in an institution, the number of learners increases until saturation; hence the following algorithm:

Local: E school

a learner to register

i variable to count the number of registered learners

Ni the ideal number of learners in E

Nc the current number of learners in E

1. **Begin**

2. if (Nc <Ni) then

3. if (Enroll (E, a)) then

4. Nc = Nc + 1

5. endif

6. endif

7. **End**

5.3.6 The function "decrease"

Whenever a learner is dismissed from school or no longer enrolls for a variety of reasons, he releases a seat. For this purpose the school number of learners is then reduced and another learner is eligible for enrollment in the class of the latter. The algorithm is as follows:

Local: E school

a learner to register

Nc the current number of learners in E

Function dismissal which tests if learner is dismissed

abandon which tests if learner has abandoned the school E

1.**Begin**

2. if (dismissal (E, a)) or abandon (E, a) then

3. Nc = Nc-1

4. Endif

5. **End**

5.3.7 The function "build a school"

For the construction of a school, when all the resources are available, it is necessary to choose a locality where to build the school. The algorithms of the choice or not of the locality where to build a school are as follows:

For primary schools

Variables declaration:

E, L: String

eDEU, eDER, eEE, ePNE, and eZU, eZR, eFER, eFEU, eEXSIL: boolean

eEXSIL, eEXCP, eEXCE1, eEXCE2, eEXCM1, eEXCM2, E1, E2: boolean

Role of variables:

E school

L Locality

eDEU indicates whether the distance from urban school to learner location is acceptable

eDER indicates if distance from rural primary school to learner location is acceptable

eEE indicates if teachers are available for school

ePNE indicates if non-teaching staff is available for school

eT indicates if the relief where the school will be built is accessible

eZU indicates if it is an urban area

eZR indicates if it is a rural area

eFER indicates if the finances available to build a school in rural areas are sufficient

eFEU indicates if the finances available to build a school in urban areas are sufficient

eEXSIL indicates if the number of students who cannot enroll in class one is greater than 0 in the locality

eEXCP indicates if the number of students who do not have the opportunity to register for class two is greater than 0 in the locality

eEXCE1 indicates if the number of students who do not have the opportunity to enroll in class three is greater than 0 in the locality

eEXCE2 indicates if the number of students who cannot enroll in class four is greater than 0 in the locality

eEXCM1 indicates if the number of students who are not able to enroll in class five is greater than 0 in the locality

eEXCM2 indicates if the number of students who cannot enroll in class six is greater than 0 in the locality

E1: Boolean variable combining the conditions on the material, financial and human resources necessary for the construction of a school mentioned above

E2: Boolean variable combining conditions on pupil numbers by class of schools

Function:

Build (E, L) which makes it possible to build a school E on a locality L

Specifications:

eDEU = 1 if the distance from urban school to learner location is acceptable and eDEU = 0 otherwise

eT = 1 if relief where to build is accessible and eT = 0 otherwise

eZR = 1 if the locality for which the school will be built is rural and eZR = 0 otherwise

eZU = 1 if the locality is urban and eZU = 0 otherwise

eEE = 1 if availability of teachers for school and eEE = 0 otherwise

ePNE = 1 if availability of non-teaching staff for school and ePNE = 0 otherwise

eFER = 1 if the finances available to build the school in rural areas are sufficient and eFER = 0 otherwise

eFEU = 1 if the finances available to build a school in urban areas are sufficient and eFEU = 0 otherwise

eEXSIL = 1 if the number of students who cannot enroll in class one is greater than 0 in the locality and eEXSIL = 0 otherwise

eEXCP = 1 if the number of students who cannot register at class two is greater than 0 in the locality and eEXCP = 0 otherwise

eEXCE1 = 1 if the number of students who cannot enroll in class three is greater than 0 in the locality and eEXCE1 = 0 otherwise

eEXCE2 = 1 if the number of students who cannot enroll in class four is greater than 0 in the locality and eEXCE2 = 0 otherwise

eEXCM1 = 1 if the number of students who cannot enroll in class five is greater than 0 in the locality and eEXCM1 = 0 otherwise

eEXCM2 = 1 if the number of students who cannot enroll in class six is greater than 0 in the locality and eEXCM2 = 0 otherwise

Let $E1 = (eDEU \wedge eT \wedge eFEU \wedge eZU \wedge eEE \wedge ePNE) \vee (eDER \wedge eT \wedge eFer \wedge eZR \wedge eEE \wedge ePNE)$

E1 = 1 if eDEU = 1 and eT = 1 and eFEU = 1 and eZU = 1 and eEE = 1 and ePNE = 1 or eDER = 1 and eT = 1 and eFER = 1 and eZR = 1 and eEE = 1 and ePNE = 1 ;

E1 = 0 if eDEU = 0 or eT = 0 or eFEU = 0 or eZU = 0 or eEE = 0 or ePNE = 0 and eDER = 0 or eT = 0 or eFER = 0 or eZU = 0 or eEE = 0 or ePNE = 0

Let E2 = eEXSIL ∨ eEXCP ∨ eEXCE1 ∨ eEXCE2 ∨ eEXCM1 ∨ eEXCM2

E2 = 1 if eEXSIL = 1 or eEXCP = 1 or eEXCE1 = 1 or eEXCE2 = 1 or eEXCM1 = 1 or eEXCM2 = 1

E2 = 0 otherwise

Begin

If (E1 = 1 and E2 = 1) then

Build (E, L)

End

- For high schools

Variables declaration:

Ly, L: String

eDLU, eDLR, eEL, ePNL, eT, eZU, eZR, eFLR, eFLU, eEXSIX: boolean

eEXCINQ, eEXQUAT, eEXTROIS, eEXSEC, eEXPRE, eEXTER, E3, E4: boolean

Role of variables:

Ly: High School

L: Locality

eDLU: indicates if the distance between the place of construction of the urban High School and the learner location is respected

eDLR: indicates if the distance between the place of construction of the rural High school and the learner location is respected

eEL: indicates if teachers are available for high school to build

ePNL: indicates if non-teaching staff is available for high school to build

eFLR: indicates if the finances available for the construction of the high school in rural area are sufficient

eFLU: indicate if the finances available for the construction of a high school in urban area are sufficient

eT indicates if the relief where the high school will be built is accessible

eZU indicates if it is an urban area

eZR indicates if it is a rural area

eEXSIX: indicates if the number of students who cannot enroll in From one in the locality is greater than 0

eEXCINQ: indicates if the number of students who do not have the opportunity to enroll in From Two in the locality is greater than 0

eEXQUAT: indicates if the number of students who cannot enroll in the From Three in the locality is greater than 0

eEXTROIS: indicates if the number of students who cannot enroll in From Four in the locality is greater than 0

eEXSEC: indicates if the number of students who cannot enroll in From Five in the locality is greater than 0

eEXPRE: indicates if the number of students who cannot enroll in From Six in the locality is greater than 0

eEXTER: indicates if the number of students who cannot enroll in From Five in the locality is greater than 0

E3: Boolean variable combining the conditions on the material, financial and human resources necessary for the construction of a high school mentioned above

E4: Boolean variable combining conditions on student enrollment in high school class

Function:

Build (Ly, L) that allows to build a high school Ly on a locality L

Specifications:

eDLU = 1 if the distance from urban high school to learner location is acceptable and eDLU = 0 otherwise

eDLR = 1 if the distance from rural high school to learner location is acceptable and eDLR = 0 otherwise

eT = 1 if relief where to build the school is accessible and eT = 0 otherwise

eZR = 1 if the locality for which the school will be built is rural and eZR = 0 otherwise

eZU = 1 if the locality for which the school will be built is urban and eZU = 0 otherwise

eEL = 1 if availability of teachers for high school and eEL = 0 otherwise

ePNL = 1 if availability of non-teaching staff for high school and ePNL = 0 otherwise

eFLR = 1 if the finances available to build the high school in rural area are sufficient and eFLR = 0 otherwise

eFLU = 1 if the finances available to build the high school in urban areas are sufficient and eFlU = 0 otherwise

eEXSIX = 1 if the number of students who cannot enroll in From One is greater than 0 in the locality and eEXSIX = 0 otherwise

eEXCINQ = 1 if the number of students who cannot enroll in From Two is greater than 0 in the locality and eEXCINQ = 0 otherwise

eEXQUAT = 1 if the number of students who cannot enroll in From Tnree is greater than 0 in the locality and eEXQUAT = 0 otherwise

eEXTROIS = 1 if the number of students who cannot enroll in From Four is greater than 0 in the locality and eEXTROIS = 0 otherwise

eEXSEC = 1 if the number of students who cannot enroll in From Five is greater than 0 in the locality and eEXSEC = 0 otherwise

eEXPRE = 1 if the number of students who cannot enroll in

From Six is greater than 0 in the locality and eEXPRE = 0 otherwise

eEXTER = 1 if the number of students who cannot enroll in From Seven is greater than 0 in the locality and eEXTER = 0 otherwise

Let E3 = (eDLU $\wedge eT \wedge eFLU \wedge eZU \wedge eEL \wedge ePNL$) ∨ (eDLR $\wedge eT \wedge eFLR \wedge eZR \wedge eEL \wedge ePNL$)

E3 = 1 if eDLU = 1 and eT = 1 and eFLU = 1 and eZU = 1 and eEL = 1 and ePNL = 1 or eDLR = 1 and eT = 1 and eFLR = 1 and eZR = 1 and eEL = 1 and ePNL = 1;

E3 = 0 if eDLU = 0 or eT = 0 or eFLU = 0 or eZU = 0 or eEL = 0 or ePNL = 0 and eDLR = 0 or eT = 0 or eFLR = 0 or eZU = 0 or eEL = 0 or ePNL = 0

Let E4 = eEXSIX ∨ eEXCINQ ∨ eEXQUAT ∨ eEXTROIS ∨ eEXSEC ∨ eEXPRE ∨ eEXTER

E4 = 1 if eEXSIX = 1 or eEXCINQ = 1 or eEXQUAT = 1 or eEXTROIS = 1 or eEXSEC = 1 or eEXPRE = 1 or eEXTER = 1

E4 = 0 otherwise

Begin

If (E3 = 1 and E4 = 1) then

Build (Ly, L)

End

- For universities

Variables declaration:

L, A: String

eDUUn, eDRUn, eT, eZU, eZR, eEUn, ePNUn: Boolean

eFRUn, eFUUn, ePUn, eFe-LEARN, E6, E5, eE-LEARN: Boolean

Role of the variables:

eDRUn: indicates if the distance from the construction site of rural university to the learner location is respected

eDUUn: indicates if the distance separating the rural university construction site to the learner urban location is respected

eT indicates if the relief where the university will be built is accessible

eZU indicates if it is an urban area

eZR indicates if it is a rural area

eEUn: indicates if there is availability of teachers for the university

ePNUn: indicates if there is availability of non-teaching staff for the university

eFRUn: indicates whether the finances available to build a university in rural areas are sufficient

eFUUn: indicates whether the finances needed to build a university in urban areas are sufficient

ePUn: indicates whether the number of people seeking university enrollment in the locality who do not have the opportunity due to lack of university is greater than 0

E5: Boolean variable verifying the conditions on the material, financial and human resources needed for the construction of a university

E6: Boolean variable verifying conditions on student enrollment in university.

Function:

Build (Un, L) that allows to build a university on a locality L

Specifications

eDRUn = 1 if the distance from the rural university to the learner location is acceptable and eDRUn = 0 otherwise

eDUUn = 1 if the distance between the urban university location and the learner locality is acceptable and eDUUn = 0 otherwise

eEUn = 1 if availability of teachers for university and eEU = 0 otherwise

ePNUn = 1 if availability of non-teaching staff of the university and ePNUn = 0 otherwise

eFUUn = 1 if the available finances are sufficient for the construction of a university in urban area and eFUUn = 0 otherwise

eFRUn = 1 if the available finances are sufficient for the construction of a university in rural area and eFRUn = 0

ePUn = 1 if the number of students who cannot enroll at the university is greater than 0 in the locality and ePUn = 0 otherwise

eT = 1 if relief where to build is accessible and eT = 0 otherwise

eZR = 1 if the locality for which the university will be built is rural and eZR = 0 otherwise

eZU = 1 if the locality is urban and eZU = 0 otherwise

E5 = (eDUUn$\wedge eT \wedge eZU \wedge eEUn \wedge ePNUn \wedge eFUUn$) ∨ (eDRUn$\wedge$ eT $\wedge eZR \wedge eEUn \wedge e\,PNUn \wedge eFRUn$)

E5 = 1 if eDUUn = 1 and eT = 1 and eFUUn = 1 and ePNUn = 1 and eZU = 1 and eEUn = 1 or if eDRUn = 1 and eT = 1 and eFRUn = 1 and ePNUn = 1 and eEUn = 1 and eZR = 1

E5 = 0 otherwise

Let E6 = ePUn

E6 = 1 if ePUn = 1

E6 = 0 otherwise

Begin

If (E5 = 1 and E6 = 1) then

Build (Un, L)

End

The following part will allow us to present Multi-Agent Simulation for the Construction of a High School

6. MULTI-AGENT SIMULATION FOR THE CONSTRUCTION OF A HIGH SCHOOL

The GAMA simulation software platform is used. It is developed under GPL / LGPL license and it is free [8]. It integrates a comprehensive modeling language (GAML) and an integrated development environment: this allows modelers (even non-computer scientists) to build models quickly and easily. GAMA is developed in JAVA and is easy to expand to take into account specific needs. It integrates tools to analyze the models including parameters of exploration of the space and the calibration of the models [8].

6.1 Highlights of GAMA compared to other simulation platforms

Gama has the following strengths compared to other simulation platforms [8]:

Gama supports the development of fairly complex models;

It seamlessly integrates geographic data and GIS tools with agent-based models;

It integrates a methodological approach to define multi-level models;

It integrates high-level tools: multi-criteria decision-making tools, grouping functions, statistical operators...

It is easily extensible thanks to its open architecture, which is based on two legacy Java technologies: the OSGI plugin framework and Java annotations.

6.2 Presentation of the simulation

A shapefile from high schools of Mfoundi department is used to realize our simulation. High schools have a green color while learners looking for an enrollment in a high school are black. Initially we have learners without schools; these will move on a radius of 3.5 km maximum; if there are places for their levels they can enroll. In the case where there are several students without school after this quest then the person in charge of the construction schools is interpellated. Figure 3 shows the Mfoundi department with 500 learners looking for enrollment.

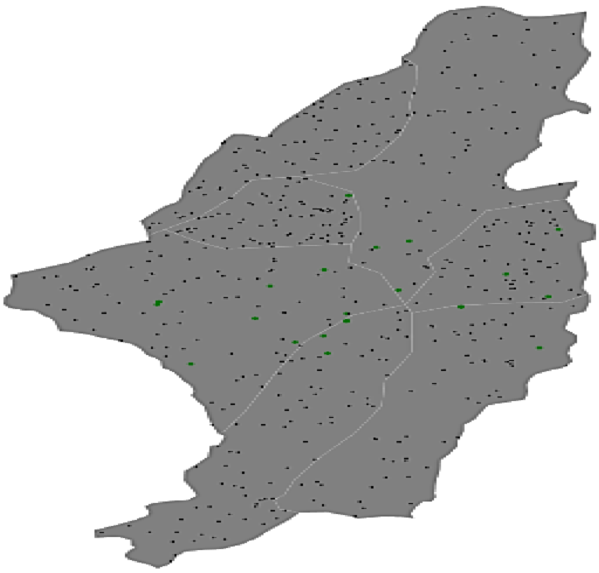

Fig 3: Initial situation where learners are looking for high schools to register

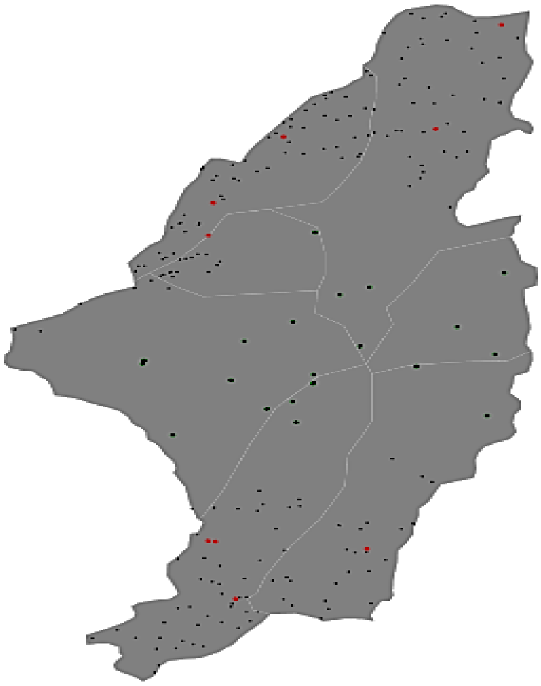

Fig 4: proposal solution situation for the construction of high schools

When the search for a school is launched for an enrollment, depending on the availability of places the situation of Figure 4 is obtained

This situation shows that there are 197 students who have not been able to enroll and 303 only who have taken an enrollment. The simulation also proposes where to build new establishments that are here in red

7. CONCLUSION AND PERSPECTIVES

At the end of our work, where it was talking about setting up a multi-agent decision-making model for the construction of a school, the following terms were been defined: decision support, multi-agent systems and agent. A classification of the different types of agents and a comparison of the agents and objects concepts were been also presented. After that, the different forms of interactions between the agents were been showed.

The following allowed highlighting of all agents of the model as well as an architecture of the latter, then the different functions for each agent were presented and theirs implementations in algorithm form. Finally, an example of multi-agents simulation of high school construction with GAMA platform was presented.

This model is thus obtained from the various agents identified and different interactions between them presented in the architecture as well as different implementations of the functions performed by each agent.

In perspective there is a plan to set up a model for building classrooms in high schools.

8. REFERENCES

[1] Direction of Construction and Heritage of the Ministry of Education, Higher Education, Training of Executives and Scientific Research of the Kingdom of Morocco. 2009. School Construction Design, Nomes and Directives, Volume 2 Collegial Secondary Education and Qualifier

[2] Division of School Engineering of the Ministry of National Education and Vocational Training. June 2013. School Building Standards

[3] Dominique URBANI. 2006 «Development of a hybrid MAS-GIS approach for the definition of a decision support system; application to water management ", PhD thesis of the University of Corsica - PASQUALE PAOLI.

[4] Roy B. 2000 "Reflections on the theme: Quest for the optimum and help with decision." In Decision, Prospective Auto-Organization.. Dunod (Eds.).

[5] Ferber J. 1995 "Multi-Agent Systems, Towards Collective Intelligence", InterEditions,.

[6] Russell S.J. and Norvig P. 2003. "Artificial Intelligence: A Modern Approach". Pearson Education.

[7] Wooldridge M. 2001 "An introduction to multi-agent systems", editions John Whiley & Sons.

[8] Alexis Drogoul, Benoit Gaudou, Patrick Taillandier, Philippe Caillou, Arnaud Grignard and Nicolas Marilleau, Nov. 2012. Course 1c: Introduction to Gama from GAMA, Can Tho University / IRD

[9] Peter Habermehl. 2005-2006. "Artificial Intelligence Course",

[10] Kele Mbbe R. C., Nkenlifack, M., Kamla V. 2019. Multi-level and generic decisional model for e-governance applications in the educational field.