Distributed Individual-Based Environmental Simulation

Mark Scahill,
Computing Laboratory, University of Kent at Canterbury.
The University, Canterbury, Kent, CT2 7NF, UK.
Telephone: +44 1227 823979. Fax: +44 1227 762811
e-mail: M.Scahill@ukc.ac.uk

Abstract

This paper describes the development and construction of a distributed model allowing the simulation of a large population. Particular attention will be paid to allowing the modelling of an individual's behaviour, communication and interaction with a shared environment. Individual based modelling is not a new concept, nor is the idea of distributed simulations, the system detailed here offers a means of combining these two paradigms into one large-scale modelling environment.

A key concept in this system is that each individual being modelled is implemented as a separate process. This atomisation of the model allows the simulation a greater flexibility, individuals can be rapidly developed and the simulation can be spread over a wide number of machines of varying architectures.

In an attempt to produce a flexible, extensible, individual based model of a large number of individual subjects the client-server paradigm has been employed. Combining the individual-based modelling techniques with a client-server network architecture has been found to be quite straightforward with the added bonus of having communication between individuals included for free. The idea of considering the problem as one of interaction between an individual and the environment means that the problems normally associated with distributed simulations, those of continuity of world-views for different clients and of communication between clients, are easily solved.

Although this system has been developed originally to allow simulations of the Mountain Gorilla (Gorilla Gorilla Beringe) population, the modelling methods employed have meant that almost any entity can be simulated with very little change to the basic simulation processes.

Keywords

Distributed, Individual-based, Population Modelling.

1. INTRODUCTION

This paper describes the construction of a distributed individual-based model aimed, primarily, at the simulation of the Mountain Gorilla (Gorilla Gorilla Beringe) population. Areas which are to be studied will include: gorilla behaviour, population dynamics and environmental dependencies.

Mountain Gorillas were chosen for this case study for a number of reasons:

- They form stable social groups, leading to relatively simple rules of interaction.
- The environment they live in has been documented in great detail over the past thirty years.
- The population currently numbers less than seven hundred individuals.
- Their habitat is a closed geographical area.
- They are officially an endangered species so any information about the future survival of the species could help present preservation planning.

In this simulation each gorilla will be modelled as a separate individual with information being passed between a gorilla and an environment which is shared between all the gorillas being modelled. This sort of communication between individuals and the environment has a number of advantages:

- The model can be implemented over a network of computers which can prevent any one computer from becoming too heavily loaded.
- Each gorilla can have their own view of the world, independent from other gorillas, yet appear in other gorillas' views.
- Gorillas may learn and act on an individual basis whilst being aware of the actions of other gorillas in their neighbourhood.
- Interactions between gorillas can be easily implemented.

Effectively, 'gorillas' can be of different species without any modification to the basic simulation.

The environment is modelled as a rectangular grid containing information on the height of each grid-point, the vegetation type and any activity in the area. The environmental process accepts actions from all gorillas and then sends back to individuals the results of their actions and the new - updated - view that the individual may have of its own locality. For example, it could incorporate the results of actions by other gorillas in the neighbourhood.

For the rest of this paper the individuals being modelled will be referred to as 'gorillas' for ease of reading. The simulator has been developed to be independent of the species being studied.

2. INTERACTION

In any investigation of the behaviour of a number of individuals, it is important to allow those individuals an ability to interact with each other. The use of a common environment in the system provides a reliable means of ensuring all interactions between individuals can be experienced by individuals in the locality as well as those directly involved in the interaction.

Communication between gorillas includes visual, audio and olfactory senses. Visual expressions are easy to model because gestures can be directed at individual gorillas although the gorilla gesturing may be observed by other gorillas. Sounds may also be directed at another gorilla, but in this case, gorillas unable to see the source of the sounds would not know which gorilla the sound was aimed at or, to some extent, which gorilla made the sound. Alarm sounds would therefore be reacted to by all gorillas able to hear the alarm just in case the warning is intended for them. Smells are not used in conscious communication. A gorilla would be able to smell other gorillas, ideally recognising the smell, over a certain distance (depending upon the weather and vegetation) and gain an idea of which gorillas lie in which direction.

The simulation environment must allow for these various interaction media and be able to broadcast actions over a varying range depending upon the nature of the action. Figure 1 shows the range of these senses. A given gorilla may see the events occurring in the cell they currently reside in and each of their neighbouring cells. They may hear any sounds from these cells and cells one step further out. Finally (depending upon wind speed and direction) they can detect odours up to three cells out from their present location. The nearer a sound or smell originates, the stronger the experience.

The remaining form of interaction, that of physical contact, can only be experienced by gorillas located in the same cell at the same time. These interactions are directly experienced by one other gorilla, for example the gorilla being groomed, but my be observed by neighbours.

To model the behavioural aspects, gorillas will have ‘relationships’ with other gorillas. New-born gorillas will have a direct dependency with their mother and mothers will have strong maternal feelings towards their baby. Both these needs will decrease as time passes and the infant matures. Other gorillas will have similar links with each other. Where one gorilla has a strong link with a more senior gorilla they will follow that gorilla. Eventually there will be one gorilla who feels no need to follow any other gorillas, this will be recognised as the group leader. Male gorillas should exhibit some rivalry, especially between male gorillas of different social groups. Where one group has two male gorillas of similar age and size it could be possible that the loyalties of the group members would be split between these two potential leaders. In this situation the protective roles of the adult males of the group could lead to some resolution of leadership with one gorilla ousting the other.

Each gorilla then has a number of factors which decide upon how much they like or dislike another gorilla. These factors form the basis of social organisation of the population. Schaller (1963) described the bonds which kept gorilla groups together yet limited their size to between five and thirty individuals.

These Factors include: the relationships between parents and offspring, which is more pronounced in females; the protection given to young gorillas by more senior members of the group; the loyalty or respect gorillas have to their senior members; the rivalry between gorillas of similar stature and the intimidation subordinate gorillas may feel about larger more aggressive individuals.

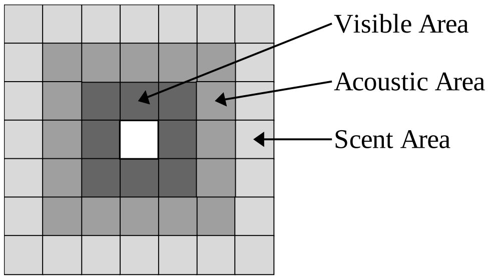

Figure 1: Gorilla sensory zones

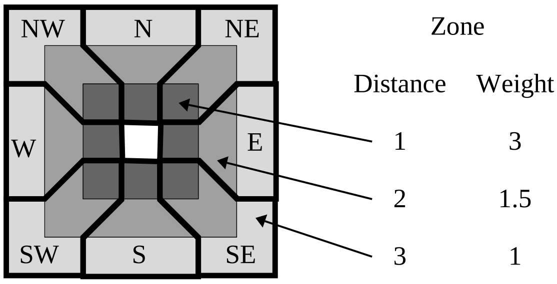

Figure 2: Gorilla movement zones

In general, the forces acting between gorillas will be a combination of these factors. The attractive forces acting to keep groups together and the repulsive forces helping to stop all the gorillas from trying to be in the same group.

These inter-gorilla relationships are planned to be modelled as a set of **springs** combining the attractive and repulsive forces listed above. A gorilla will be governed by its desires to be near some gorillas and away from others, in a certain vegetation type or doing a certain activity. This conflict of interests will be the basis for group formation and stability.

3. THE SIMULATION

In this system, each individual is modelled as a separate process and the environment as another process. This atomisation of the model enables rapid development of the system whilst enabling individuals the ability to easily modify their own world views and behaviour. The alternative method of modelling large numbers of individuals on this scale would be to have one large monolithic system with all control being handled internally. The complications of ensuring separate world views for each gorilla in this kind of system and allowing individual's the ability to develop their own behaviour would create a massive and unwieldy system. By dividing the model into a large number of smaller processes each individual process requires less resources from the host computer and, as mentioned, allows gorillas to be distributed over a network of computers reducing the demand (in terms of processor time and system memory) on any one machine.

The simulation is performed through a series of discrete time steps. During any given time step each gorilla will decide upon its action and obtain the results of those actions. Possible actions include movement, feeding, interacting with other gorillas and sleeping.

Each gorilla remembers areas of the environment which it has experienced. When deciding upon a given movement step a gorilla will consider its immediate surroundings and where it remembers a given resource (such as a food-type) to be obtainable. An individual gorillas movement is based on the 49 grid cells surrounding their current location (3 cells in each direction as indicated in Figure 1). Decisions about which of the immediately adjoining cells to move to will be made based upon what they can sense in those cells, their memory of the location/area if they have been there before and any environmental feedback from the surroundings.

Environmental feedback may well come from up to three squares away (in the case of smells) but the gorilla experiencing the smell can have no idea how far away the smell originated, only the direction could be determined. A similar argument applies to sounds so the sensory feedback is going to appear to have come from the nine cells adjacent to the gorillas present location. These nine cells, the currently occupied cell and its eight adjoining cells are all that need to be passed back from the environment to a gorilla.

Based on these senses, a gorilla will decide to move in a given direction. There are nine sensory areas: N, NE, E, etc. and stationary. The stationary zone is the current gorilla location, the other zones comprise of an area of six grid squares each. Figure 2 shows how these zones are arranged. Each zone contains the same number of cells, half cells being shared equally between the two neighbouring zones, and the same number of cells at each distance.

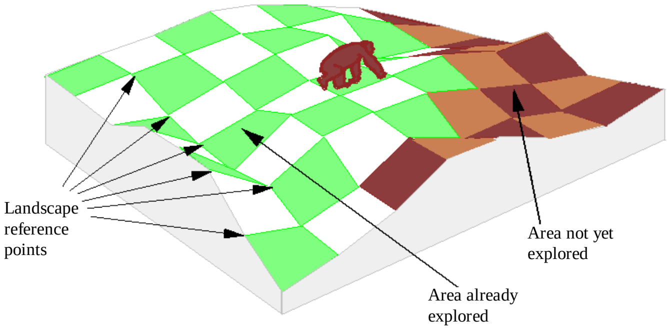

Figure 3: Simplified view of the landscape

In considering which direction to move, an individual considers the contents of each cell in the zone with a weight depending upon how far away the cell is. In this way, the three farthest cells in a particular zone will have the same combined importance as the two intermediate cells, which will in turn have the same importance as the immediate cell.

Also to be considered are land heights and vegetation types. Figure 3 shows how the landscape might be represented as a series of grid locations with associated heights. Heights are important since different vegetation types tend to be found at certain altitudes. As gorillas learn more about their environment they will learn to associate sources of food with different altitudes. Each reference point in the landscape contains information on the altitude and vegetation type at that point. This information is available for the whole of the Virungan Volcano region where the mountain gorillas were first found. The other area where mountain gorillas can be found, the Bwindi Forest Reserve in Uganda, is currently being studied though being geographically separate from the Virungan region does not pose a problem.

Although the actual landscape being modelled is not as regular as this representation implies, the abstraction makes the model more simple to construct without being too artificial. The data for the landscape is mostly obtained from the Grant F. Walton Center for Remote Sensing and Spatial Analysis, where the elevation map was generated from Belgium maps and the vegetation from a mixture of Satellite photographs and ground based observations.

4. COMMUNICATION

Communication in this simulation is performed through the use of Unix TCP sockets. This means that two processes can set up a link between each other and then pass information along this link. The communications link will be private to each process and allow information to travel in both directions. Using Unix sockets means that these links can be between processes on the same physical machine, or between processes on different machines. As far as the system is concerned, all the complicated message passing routines are hidden by calls to the operating system which send information to and receive information from a message channel.

Early on in the development of this modelling system, it became apparent that modelling over three hundred individuals on the same computer was not going to be feasible. A limit on the number of channels that could be open at one time meant that it was not possible to model more than about forty gorillas simultaneously on any one computer.

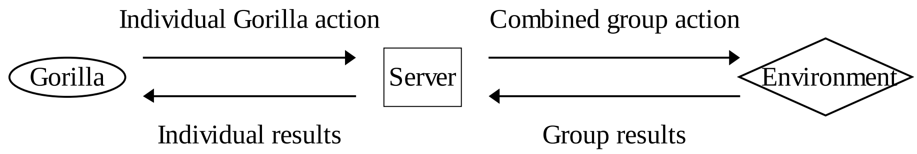

Figure 4: gorilla to environment communication.

Since the limiting factor was the number of channels, spreading the gorillas out over a number of machines would not have solved this problem as each gorilla would still have to communicate with one single environment. This problem was solved by giving each machine a server which acted as an intermediate service between the gorilla and the environment, as shown in Figure 4. With the servers method, the environment would only need to communicate to twenty other processes, these being the group servers and each server would only need to communicate with around thirty other processes, namely the gorillas and one link to the environment. Therefore a simulation of the entire mountain gorilla population of just over 600 is possible using the servers as an intermediary between the environment and the individual gorillas. Each machine running gorilla processes would require only one server and thus only one link to the central environment.

By spreading the simulation over a number of machines there is also the bonus of allowing gorillas to process the results of their actions in parallel. This allows a substantial increase in the number of actions which can be performed in a given time over the single modelling process normally used for individual-based simulation. Actions are passed to the group server which in turn passes them on to the environment. The environment processes these actions and sends the results back to the gorilla, via the group server.

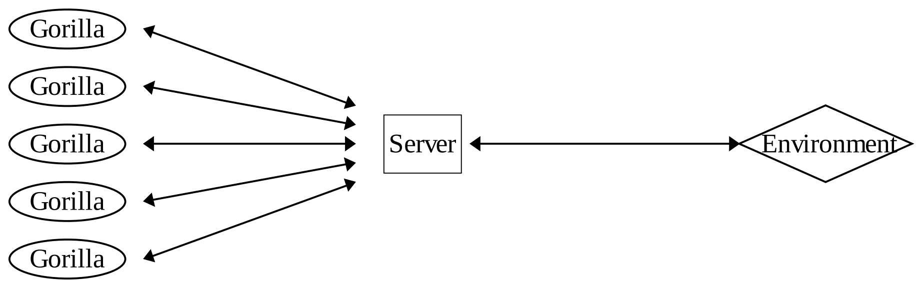

Figure 5 Group Communication

As shown in Figure 5, the server combines the actions of its associated gorillas and pass the groups actions on to the environment. The environment passes back the combined results to the server which divides these combined actions up and distributes individual results to the relevant gorilla.

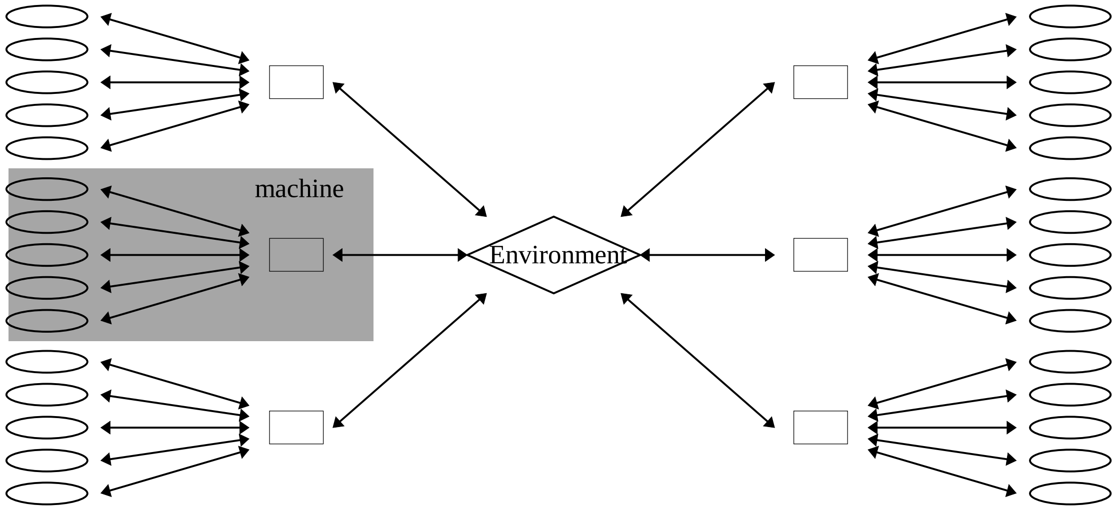

Figure 6: Environment Communication

Figure 6 shows the environment’s view of a system of thirty individuals on six separate machines (seven including the environment). The environment cycles though the servers receiving all the actions for each collection of individuals, then processes all the actions and sends results back to each server in turn. The environment, once it has received all the servers' actions, will treat gorilla actions on an individual basis. In general, individuals actions are not effected by being on a specific machine.

The communication arrangement of the system allows each gorilla to share a common environment, communicate with other gorillas in their vicinity and allow individual gorillas to form their own view of this environment. Being able to spread the gorillas out among other machines prevents any one machine from becoming too loaded and the servers, acting as multiplexors, reduce the number of network connections and the network lag which would have been present had each gorilla communicated directly with the environment. These factors combined result in the system performing much faster with a larger number of gorillas than it did before the servers were introduced and the gorillas distributed over a number of networked machines.

5. COMMUNICATION PROTOCOLS

Since the most important part of this system is the communications between processes it is important that enough information is shared between individuals and the environment to allow individuals to make valid decisions on their actions and to interact as naturally as possible with each other.

A typical action is sent as a series of twelve bytes, as shown in Figure 7. These packets contain details of the individuals unique ID number, the group-server the gorilla belongs to (for authentication purposes) the size of the individual and their location. It then contains their action (e.g. Moving, Eating, grooming, etc.), the intention behind the action (anger, peaceful etc.) and the ID of the individual the action is aimed at. The server combines all these actions together into a larger packet, adds the server id (for authentication) and the number of gorillas which the server is responsible for serving.

0	Ape ID		Server ID	Size	
4	X-coord		Y-coord		
8	Action	Intent	To Ape		12

Figure 7: Gorilla to server packet

The server then combines the actions received from its client individuals and combines them into a single packet to be passed on to the environment. Figure 9 shows this packet which contains a four byte header plus twelve bytes per individual served. For a group serving thirty individuals this packet will be 364 bytes long.

0	Server ID	Apes in Group	
4	Ape 1 action		
16	Ape 2 action		
28	Ape 3 action		
40	Ape 4 action		52

Figure 9: Server to environment packet

Upon receiving all the servers' packets the environment performs the actions for each gorilla and calculates the results. These results are then compiled into a packet containing the results for all the individuals catered for by a given server and sent to that server. The environment to server packet will be these packets together with the server's ID number and the number of gorillas being served, as shown in Figure 8, which may be more than the group previously served in the case of new gorillas being introduced to the system (through births or migration).

0	Server ID	Apes in Group	4
	Ape 1 results		
	Ape 2 results		
	Ape 3 results		
	Ape 4 results		

Figure 8: Environment to server packet

The packet sent by the server back to an individual, shown in
Figure 10, contains the individuals ID number followed by the vegetation and heights of the adjacent nine grid locations. This landscape information comprises eighteen bytes (nine each for vegetation and height). The remainder of the packet is made up of sounds, smells and the actions of neighbouring gorillas. There are nine bytes of sounds and smells, one byte for each grid location (direction) which the sounds or smell appears to be coming from. Here the type of sound (anger, fear, peaceful) and the type of smell (fear, food etc.) is stored as a binary bitmap. This bitmap is one byte long with each of the eight bits describing a sound or smell. The first four bits are sound types the last four are smell types. This could be extended to allow gorillas the ability to recognise the scent and vocalisations of other gorillas. The number of neighbouring gorillas then follows (2 bytes) along with the observed actions of

0	Ape ID
2	Vegetation
11	Heights
20	Sounds + smells
29	Number of local apes
31	Neighbour 1 Action
39	Neighbour 2 Action
47	Neighbour 3 Action
55	

Figure 10: Server to Gorilla Packet

each neighbour: the neighbours ID (2 bytes), their size (1 byte), their location relative to the destination gorilla (1 byte), their action (1 byte), their intent (1 byte) and which gorilla the action appears to be aimed at (2 bytes) (essentially, the same as the action the neighbouring gorilla sent to the environment without the server id. and the neighbours location made relative to the target gorilla's location. For a gorilla with no neighbours, the results packet will be 31 bytes long whilst a gorilla with ten neighbours will receive a packet 111 bytes long.

This means that for a server with eleven distributed individuals, the result packet received from the environment will be 341 bytes, whilst a group with eleven gorillas within close proximity to each other will contain 1221 bytes (each gorilla being informed of its ten neighbours actions).

The system for sending packets over the network (the Unix send and receive commands) allows the receiver of a packet to check how many bytes it has received. This allows the integrity of the data to be checked by all parties on the network should this be required. It also provides a simple method of communication whilst hiding the actual message passing protocol from the components of the simulation.

Table 1 shows the volume of network traffic created for one server when serving thirty individuals in close proximity to each other.

Conversation	Packet		bytes sent	Sent Over
	Size	Number	each time step	
Gorilla to Server	12	One Per Gorilla	360	Local
Server to Environment	364	One Per Server	364	Network
Environment to Server	7894	One Per Server	7894	Network
Server to Gorilla	263	One Per Gorilla	7890	Local

Table 1: Packet sizes (in bytes) for thirty gorillas in close proximity

With a simulation of 300 gorillas divided into ten groups, this means that about 80kb of information may be sent over the network for each time step (not including any protocol headers added by the system). Without the use of high-speed networks (capable of transferring over 1Mb of data in a second) this system would prove far to unwieldy because of the communication overheads involved. With these facilities available the system works quite well (though could be considered a little selfish in terms of resource usage).

In this limited set-up, eighty gorillas have been modelled at the same time, spread over five machines. In these tests actions were performed at a rate of about four gorilla actions each minute when run on Sparc Station 10s connected through 10Mb/s Ethernet.

6. FURTHER WORK

As mentioned earlier, gorillas will learn about their environment and remember where they have been (in terms of the vegetation and terrain). They will also learn how to interact with the environment and with other gorillas. Gorillas will have the ability to learn, adapting their behaviour through experience and knowledge. In terms of food, gorillas will be able to associate a given amount of food available with a certain type of vegetation in a fixed location. It is hoped that after a while they will be able to anticipate that a given vegetation type will give a predictable supply of food.

Interactions with other gorillas (and at a later stage, other species, including traps, poachers and other wildlife) will be based on expected outcomes. Using the techniques described in Steinhauer's book13, all actions will have an expected outcome. When the actual result of an action is as was expected, the experience of that action is increased. When an outcome is much worse than expected the desire to perform the action again is suppressed and when an outcome is unexpectedly good, confidence in the action is boosted.

This reward-based behaviour should be straightforward to implement and gorillas can associate actions with individual gorillas, groups of gorillas and other species. At present, the behaviour of a gorilla has not been implemented beyond the stage of looking for food on a completely selfish basis.

The model has been designed to be easily extended. Poachers will be added as mere adaptations of the gorilla code, new interaction methods will be added for hunters, traps and tourists. Although gorillas are not troubled much by other animals, the pressures on the local environment are such that the size of the national park where they live is being eroded as human activities encroach on their habitat. It is possible that the gorillas may have to react to competition for food with other species in the environment.

The simulation described in this paper is not yet complete. At the time of writing more work needs to be done on the gorilla processes, the behavioural aspects of the model detailed above need to be implemented and tested. The client-server system and the environmental process are implemented and have been found to be working.

A demonstration of the networked simulation has been built using a rudimentary sheep and sheepdog model, where a human operator is able to guide a sheepdog interactively and round up sheep into a pen. This simpler model was used to test the message passing protocols and the performance of the network simulation. The sheep were slightly modified gorillas who searched for food whilst keeping an eye on the sheepdog. If the sheepdog came too close the sheep would either run away from the sheepdog or towards the nearest neighbouring sheep. In this way the sheepdog could be positioned to keep the sheep together and gradually herd them into a pen. The environment was only changed in so far as the landscape was a field with a pen in it. Results of these tests showed the simulation to be working well with much less network delay than might have been expected; the operator was able to control the dog rounding up around thirty sheep spread over two machines without any trouble or noticeable delay.

Work to be done includes a more thorough implementation of the gorilla process. At present gorillas search for food with little attention being paid towards each other, behaviour is fixed at the start and apart from discovering the environment a gorilla has no capacity for learning. The next stage is to make gorillas more aware of each other and allow them some patterns of behaviour. It is envisaged that these behavioural patterns should be self-taught based on the actions they see other gorillas performing and the experiences they themselves have. The ability to learn about their environment, to anticipate the results of their actions and to modify their behaviour accordingly, will form a major part in the successful model of the Mountain Gorilla population simulation.

7. CONCLUSION

This paper has described the construction of a distributed simulation environment. The system has been designed with the intention of modelling an entire species numbering some six hundred individuals. Although much of the implementation has been based on the mountain gorilla population it has always been the intention that the underlying simulation methodology should be adaptable to model any species meeting the criteria of living on a closed environment, having a stable social structure and well documented patterns of interaction.

The system makes extensive use of the Unix network facilities and has been successfully implemented on a number of systems including SunOS, Ultrix and Linux. It should be easily portable to any system offering BSD compliant networking facilities. The use of Unix networking utilities enables the model to run over any TCP based network, including the Internet. No security measures have been added to the system at present making running a simulation over open network an unwise procedure.

The ability to model a large number of individuals as separate processes allows a very flexible approach to be taken in the development of this system. Other species can easily be added to the gorilla model or the system can be readily adapted to modelling completely different scenarios, as shown by the sheepdog demonstration. Immediate plans for other species include the modelling of the sheep on Soay and the red deer on Rhum where the results of these simulations can be compared with the results actual observations of their population14.

8. REFERENCES

Use the Harvard system. References should be cited in the text by name and date, e.g. (Smith, 1989) or Smith (1989). All references listed must be cited, and all cited references must be included in the reference list. Indent and second and any further lines of each reference as shown below. The list should be in alphabetical order or author; where there is more than one reference by the same author they should be listed chronologically. Give full titles of articles and journals.

Bacon, J., 1993: Concurrent Systems: An Integrated Approach to Operating Systems, Databases, and Distributed Systems, Addison Wesley

Clutton-Brock, T.H. and Price, O.F. and Albon, S.D. and Jewell, P.A., 1992:Early Development and Population Fluctuations in Soay Sheep, Journal of Animal Ecology (62/2) pages 381-396

Clutton-Brock, T.H. and Guinness, F.E. and Albon, S.D., 1984: Individuals and Populations - The Effects of Social-Behavior on Population-Dynamics in Deer, Proceedings of the Royal Society of Edinburgh Section B - Biological Sciences(82) pages 275-290

DeAngelis, D.L. and Gross, L.J., 1992: Individual Based Models and Approaches in Ecology, Chapman and Hall.

Digit News, 1992: Uganda: Mountain gorilla population doubles. Dian Fossey Gorilla Fund, Autumn edition

Fossey, Dian, 1983: Gorillas in the Mist, Houghton Mifflin Company, ISBN: 0 395 48928 8

Harcourt, A.H and Stewart, K.S. and Fossey, D., 1976: Male Emigration and Female Transfer in Wild Mountain Gorilla, Nature, 1976, 263 pages 226-227

Harcourt, A.H and Fossey, D. and Sabater-Pi, J., 1981: Demography of Gorilla gorilla, Journal of Zoology, London, 195 pages 215-233

Schaller, G.B., 1963: The Mountain Gorilla, Chicago U.P.

Schaller, G.B., 1965: The Behaviour of the Mountain Gorilla, Primate Behaviour (ed. DeVore, Irven), Holt Rinehart and Winston, pages 324-367

Steinhauer, Gene D., 1986: Artificial Behaviour, Prentice-Hall, ISBN:0-13-048844-5

Tanenbaum, A.S., 1989: Computer Networks, Prentice Hall Inc.

9. BIOGRAPHY

Mark Scahill has been a Computing Fellow at the University of Kent at Canterbury for nearly six years. When he hasn't been teaching or looking after first year exams he has been investigating the ability to model the behaviour of the Mountain Gorillas on a species wide basis. With a BSc. in Astrophysics and an MSc. in Computer Science it was either that or write space games. He has a Victorian house that still needs decorating after two years of hard work and a garden where plants die and which attracts more cats than wildlife.