This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2009 proceedings

# Effects of User Behavior on MMORPG Traffic

Géza Szabó*, András Veres*, Sándor Molnár†
*TrafficLab, Ericsson Research, Budapest, Hungary, Email: {geza.szabo, andras.veres}@ericsson.com
†High Speed Networks Laboratory, Department of Telecommunications and Mediainformatics,
Budapest University of Technology and Economics, E-mail: molnar@tmit.bme.hu

***Abstract*—Game traffic depends on two main factors, the game protocol and the gamers' behavior. Based on a few popular real-time multiplayer games, this paper investigates the latter factor showing how a set of typical game phases–e.g., player movement, changes in environment–impacts traffic. By understanding the nature of this impact an algorithm is introduced to grab specific events and states from passive traffic measurements. Further, a measurement example including a detailed analysis is shown from an operational broadband network. The collected data of player behavior in the gaming environment was analyzed and it was also shown that the long range dependence (LRD) property of gaming traffic is not due to the popular explanation based on the heavy-tailed periods of player activities.**

## I. Introduction

Several factors have been examined in current literature why network traffic of online games are such that we experience on a macroscopic level. It was found that one important aspect is its dependence on the gaming protocol itself [1], but another can be the influence of the player actions [2].

In this paper we examine how the player behavior and ingame environment affects the network characteristics of massively multiplayer online role playing games (MMORPG). It is discussed that player movement and gaming environment changes affect the traffic characteristics even on macroscopic level.

Detection of player actions would make it possible to answer such interesting questions as e.g., whether the self-similar property of human behavior appears in the traffic as well due to the player actions. We need a method to collect the necessary statistics on player actions from data sources containing bulk information. As there are a number of different games with proprietary protocols, a deep packet inspection method is not generally applicable. A statistical method would be desirable as it is general and works for even new upcoming games as well. It would make it feasible to analyze vast number of gaming sessions in non-intrusive measurements obtained at high network aggregation level. In current literature there is no such method which can collect the necessary statistics thus we propose a method which builds on heuristics to deduce the movement or idle state of the player and its surrounding area conditions from passive measurements.

We observed the internal structure of the network traffic when the player is in moving or stalling state and the environment around him when it is crowded with other players or deserted. This observation has been examined and validated with several methods and found that even if a heuristic method based on statistical properties of the traffic has several fallbacks it can be applied on the recognition of game states with high hit ratio. The proposed method was thoroughly validated on network traffic of World of Warcraft [3] and Silkroad Online [4] and was tested on Guild Wars [5], Eve Online [6] and Star Wars Galaxies [7] to check how generally applicable the method is. Although this list does not contain every possible MMORPGs but we argue that the observed traffic characteristics are common among MMORPGs, thus the proposed player state detection method may be applicable for most MMORPGs.

We examined the similarity of ingame player and real world human behavior. It cannot be expected that the two environments are completely the same, but some properties which was found earlier in connection with human activities are expected to appear in the gaming environment as well.

The rest of the paper is structured as follows: Section II collects the related work in this field. Section III introduces the methods for user behavior analysis and the validation of these methods. Section IV shows the results of the analysis of the gaming traffic of an operational broadband network. Finally, Section V concludes the paper.

## II. Related work

Today a large part of the gaming traffic is generated by MMORPGs, thus several works treating this type of traffic appeared. In [1] the authors showed that when an avatar performs different actions in the virtual world, at different places and under different network conditions an MMORPG client actually makes intensive use of network resources. They identified the reasons how the

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2009 proceedings

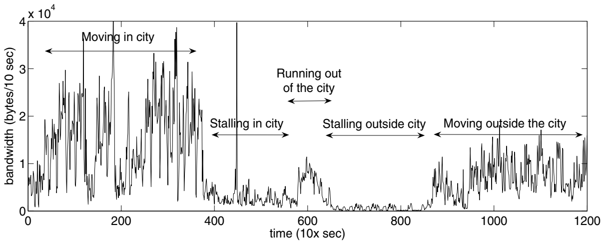



Fig. 1. World of Warcraft measurement bandwidth (bytes/10 sec)

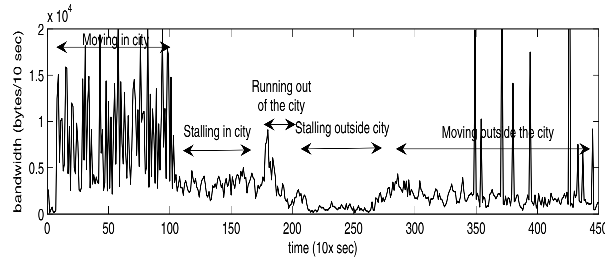



Fig. 2. Silkroad measurement bandwidth (bytes/10 sec)

high bandwidth consumption can be derived from several game servers. Authors of [2] found that the characteristic features of MMORPGs are the strong periodicity, temporal locality, irregularity, and self-similarity. The self-similarity property was explained by the existence of a hidden on-off process according to the authors. Their hypothesis was that on-off periods are due to the player active and idle periods. They defined a player active if the player sends packets above a threshold and idle if the packet sending rate of the player is below a threshold. Authors of [8] found that MMORPG bots traffic is distinguishable from human players by the regularity in the release time of client commands, the trend and magnitude of traffic burstiness in multiple time scales, and the sensitivity to network conditions.

Our work can be the missing link among the above studies. [8] provided us with the proof that human behavior is present in the network traces at macroscopic level. We argue that the self-similarity property of the gaming traffic, which was found in [2] and was modeled with an on-off model does not fit due to the impact of human behavior. In this work we show that player behavior in the MMORPG traffic has long-range dependence (LRD) properties. [2] hints that the traffic rate changes of the game may be due to player actions, but the detection heuristic has not been validated and is not precise as will be shown later in this paper. Our work extends the investigation of gaming traffic of [1] in the function of time gaining information about gaming environments, further we redefine the player activity applied in [2] to gain information about player behavior. Our work can contribute to numerous research activities not restricting for the field of network science, e.g., it makes it possible to analyze human behavior as in [9] raising several additional questions and further work.

## III. User behavior detection heuristics

In this section we introduce the methods for user behavior analysis. We constructed this method based on the analysis of active measurements.

Earlier work e.g., [2] studied game traffic characteristics with the help of power spectrum diagrams to reveal periodicity in the traffic. However the power spectrum does not grab the characteristics of the traffic in the function of time. We decided to apply wavelet analysis [10] on the MMORPG traffic to exploit the following features: one major advantage afforded by wavelets over power spectrum analysis is the ability to perform local analysis, and the other advantage is the capability of revealing aspects of data like trends, breakdown points and discontinuities in higher derivatives.

We created several active measurements in a controlled environment by capturing the generated network traffic of World of Warcraft [3] and Silkroad Online [4] and manually log the time of the different states of the player. After the measurement, we compared network characteristics and the ingame activity of the player. Examples from this experiment are shown in Figure 1 and Figure 2. The figures depict the bandwidth characteristics of the traffic coming from the server in the function of time. The two games have similar characteristics though World of Warcraft has about the twice the network bandwidth load as Silkroad. Observations on the traffic characteristics could be made in connection with player activity and the environment changes:

- *Moving/Stalling*: The player moving actions imply constant location changes, which implies constant update of the environment information around the player. The player moves on areas with refreshed state and with obsolete information, thus update information occurs intensely but with variable frequency. During stalling, the state update regularity is independent from the player and synchronized by the server with more regularity. Our observation was that the character movements increase the noise in the traffic rate time series.
- *Inside/outside city*: The density of the gaming environment of other players influences the number of independent actions which need state update at the player in their vicinity. Entering to a crowded

environment induce a level-shift upward in the traffic rate, while leaving it results in a level-shift downward.

To grab both of these characteristics in time we used wavelet decomposition (see Appendix for the Matlab [11] skeleton code of the algorithm in details):

- Reconstruction of the signal from the low frequency component gives hint about the gaming environment (densely populated city/extinct desert)
- The high frequency component gives hint about user activity (stall/run)

In Figure 3 the wavelet decomposition of the traffic presented in Figure 1 can be seen. It can be noted that the moving and stalling states are well-distinguishable on the high-frequency component: in case of moving, it becomes noisier. The environment changes can be well-tracked on the low-frequency component.

To improve the accuracy of the method we extended the wavelet decomposition with a state-machine which follows the changes in the environment. It tracks whether the player is inside city or outside city and does not let sudden changes in the environment if temporal transients occur in either of the frequency components. Figure 4 shows the defined states and state transmissions in the proposed detection algorithm. The skeleton code in the Appendix follows these states and state transitions. The high frequency component change makes a state transition between stalling and moving, while the low frequency component change results in a state transition between the environments. The environment change is only possible with the player moving. This model lacks the possible state transition between stalling in and outside the city. This could occur when a player uses teleport. In this case the state machine gets into the proper state in a few additional steps, but keeps the false state hit ratio low.

The algorithm was constructed to adapt to the analyzed traffic, thus, no fix thresholds regarding the typical bandwidth have to be set. The advantage of this feature is that any MMORPG game traffic can be analyzed without previous parameter tuning.

***Validation.*** In the validation phase 9 hours of MMORPG traffic (World of Warcraft [3] and Silkroad Online [4]) were collected and examined. The validation of the proposed method has been performed in several steps. In the first step we created several active measurements in a controlled environment by capturing the generated network traffic of the game and manually log the time of the different states of the player. After the measurement, the proposed method was applied on the network traces, and the states which were indicated by the proposed method and the logged states were compared. A state is the estimated action of the character in a given environment in a 10 sec period. In Table I it can be seen that in this phase of the validation 74% of the states were exactly recognized and 14% of the states were missed with a nearby state.

Second, we focused on Silkroad Online and World of Warcraft to check the accuracy of the method thoroughly and in bulk measurement in an automated way. Several measurements were taken in Silkroad Online and World of Warcraft when during game play, the network traffic was captured and the ingame screens were recorded into a video file. The recorded video files were processed by a heuristic algorithm to create automatically player action and game environment logs to replace the manual work for bulk measurements.

The heuristic algorithm grabs the characteristics of the ingame screens of both the environment and the moving activity. As an example outside the city, the characters march through grassy fields and forests where the over-

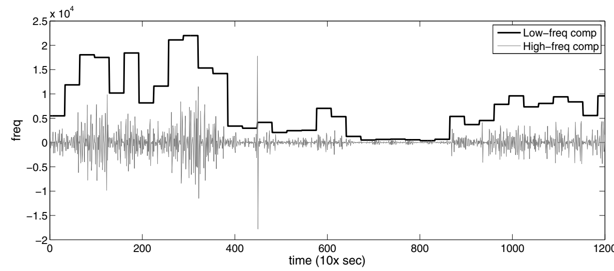



Fig. 3. Analysis of World of Warcraft traffic and the effect of different environments and character actions

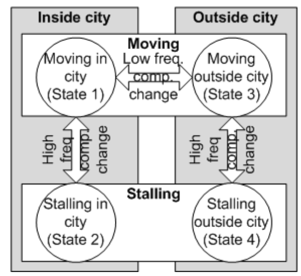



Fig. 4. States and defined state transmissions in the proposed method

| Difference | 0 | 1 | 2 | 3 |
|---|---|---|---|---|
| Manual logging | 74% | 14% | 8% | 4% |
| Video processing | 68% | 19% | 10% | 3% |

TABLE I
THE DIFFERENCE OF THE MMORPG STATE RECOGNITION ALGORITHM AND THE INGAME VIDEO PROCESSING ALGORITHM

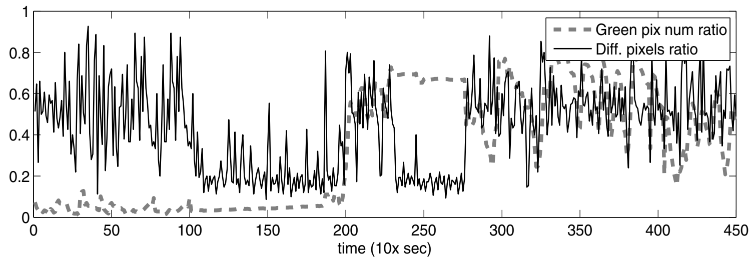



Fig. 5. Analysis of a Silkroad ingame video

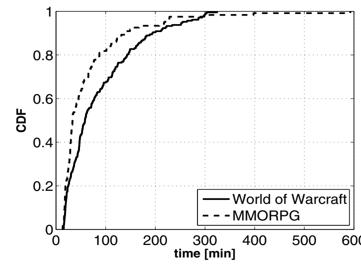



Fig. 6. Distribution of the duration of filtered MMORPGs

whelming number of the pixels has green color or at least a gradient of green. Our prototype implementation counts the number of greenish pixels of every frame of the recorded ingame video file. (We are aware of the implementation is not completely general in its current status, as e.g., a desert area where the environment is not greenish would mislead the algorithm but with a more complex color processing mechanism, our idea would be not restricted to specific gaming environments.) The ratio of the greenish pixels can be seen in the function of time in Figure 5. We used the HSL color space (hue, saturation, lightness) to check if a pixel is greenish as it makes it possible to test whether the color falls into a given hue range, but its saturation and lightness can take different values which is due to e.g., the day/night visual effect in the games. Figure 5 shows that as the character started inside the city with running around for 15 min, the green pixels ratio—the green and non-green pixels ratio on a given screenshot—is low and remains low for that period even if small fluctuations occurred. As the character stops, the green pixel ratio remains relatively constant. After the $30^{th}$ minute the player leaves the city and there is a significant rise of the green pixel ratio as in enters to the nature. The varying number of trees, grassy fields, roads, monsters, etc. results in fluctuations in the green pixel ratio, but the mean value remains high.

The player activity whether it stalls or runs is tracked by the comparison of the pixels around the legs of the player of two neighboring frames. In case of stalling, the ratio of the differing pixels is low while in the case of moving, the different moving phases result in high differing ratio. In Figure 5 it can be seen that in the first 15 minutes the character was running around, thus this ratio remained high. As it stopped the ratio dropped and the mean of this value remained low during the stalling. These two metrics make it possible two track the players' activity and gaming environments by creating simple limits. It is evident from the results that as the metrics can temporary fluctuate thus it is advisable to track their changes comparing to their sliding mean value.

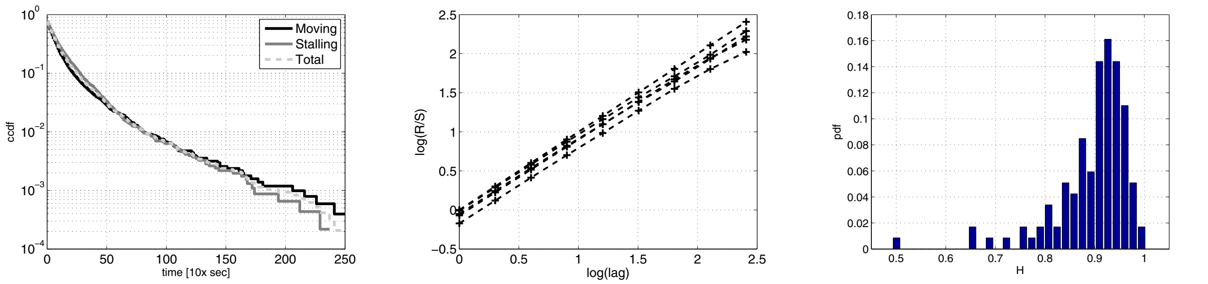
In Table I shows that 68% of the states were identified exactly and 19% of the states were missed by a nearby state. The validation showed that the algorithm misses mainly the city/outskirts transitions. It is clear that the term city is not perfectly proper as the buildings in the city and walls are just visual effects, thus the high bandwidth is in connection with the densely populated areas not the exact location if it is inside the city or not.

In the third phase of the validation process, we created several active measurement traces of the following MMORPG games beside World of Warcraft [3] and Silkroad Online [4]: Guild Wars [5], Eve Online [6] and Star Wars Galaxies [7] to check how generally applicable the method is. We found that the method performs well in that cases where the player population is high and the character actions generate considerable traffic. This states well for World of Warcraft, Silkroad Online, Star Wars Galaxies and Guild Wars as well. In the case of Guild Wars one can ask whether the method is affected by the game rule that leaving a city results in that only the members of the guild and the non-player-characters remains around the player, but the traffic characteristics does not alter considerably. Eve Online is an exception: the low rate it generates and the frequently rarely populated space territories make the algorithm inaccurate in the decision between the different states.

## IV. Analysis of Gaming Traffic in an Operational Broadband Network

Barabási [9] showed that the human activity patterns display a bursty dynamics with interevent times following a heavy-tailed distribution. Crovella [12] showed that self-similar property of network traffic is caused by user "think time". In this section we show that human behavior can result in a non-heavy-tailed but strong-correlated process causing the same LRD effect.

We measured the network traffic of an operational




(a) Sojourn time ccdf in moving or stalling periods in Wow traffic (b) R/S plots of the player states in several Wow sessions (c) Estimated H-parameter pdf of the player states in the Wow traffic

Fig. 7.

| | WoW | | MMORPG | |
|---|---|---|---|---|
| | $<$ 1h (%) | $>$ 1h (%) | $<$ 1h (%) | $>$ 1h (%) |
| City run | 23 | 20 | 19 | 19 |
| City stall | 35 | 42 | 29 | 35 |
| Out run | 30 | 27 | 25 | 30 |
| Out stall | 12 | 11 | 27 | 16 |

TABLE II
THE RATIO OF TOTAL TIME SPENT IN A SPECIFIC STATE BY THE PLAYERS

broadband network. We preprocessed the measurement with filtering out the MMORPG traffic from the total traffic with the method introduced in [13]. The filtered traffic was validated by manually checking the chat messages in the gaming sessions. Then we analyzed the filtered traffic with the methods presented in the previous sections. The examined traffic is a three-day-long passive measurement, which contained about 200 World of Warcraft flows and 100 other MMORPG flows.

In Figure 6 we can see the cumulative distribution function of the duration of filtered MMORPG traffic. It can be seen that though the average game length is about 90 minutes, we can find even 5-hour-long games.

Table II shows the ratio of time spent in different states in MMORPGs in the case of the flows less than one-hour-long and more than one-hour-long. With this differentiation it is possible to check if longer MMORPG sessions are the consequence of longer 'away from computer' status of the players. We found that the state ratio do not significantly altered which means that these players take some rest during game play but remain active in most of the time. A possible explanation of the high ratio of stalling state can be explained by the socialization activity of players according to [14] which says that a major part of the players like to use MMORPGs to chat and catch up with friends and not only play.

We intended to analyse the observed player behavior characteristics in gaming environment to reveal the possible presence of the heavy-tailed periods of human activities. Authors of [2] suggested that gaming traffic has long-range dependence (LRD) and possibly self-similar property. They constructed an on-off model for the traffic and their hypothesis was that it may fit due to the impact of human behavior with the changes of active and idle states of the players. We argue that to unfold the origin of LRD property of the traffic is a difficult and often impossible task [12]. LRD property can be the result of a superposition of heavy-tailed on-off periods [15], TCP propagation property [16], the competition of TCP sources but it may also be caused as an artifact by non-stationary.

Figure 7(a) shows the sojourn time ccdf of the moving and stalling periods in Wow traffic. It can be seen that the ccdf resembles to an exponential distribution, meaning that the distribution is not heavy-tailed. This excludes the explanation that the LRD property is the result of the heavy-tailed distributions of the movement state durations.

On the other hand, we checked the autocorrelation function of the time series of the states and it was found that the decrease of the correlation vs the lag is slow resulting in strong autocorrelation. Cross-checking this statement, we reconstructed the analysis performed in [2] by examining the time-series of the states in the function of time with the R/S plot [17] method. This time-series is such abstraction of the original traffic in which every effect – e.g., transport layer characteristics: TCP properties, packet sending rate, packet size – is eliminated and let us focus on the state change events of the player. Authors in [2] performed similar analysis except that they defined a player active if the player sends packets above a threshold and idle if the packet sending rate of the player is below a threshold. This property was found to occur due to the environment changes in our experiments. In our analysis we used our more sophisticated state definition. Figure 7(b) shows examples of the R/S plot of several WoW time-series constructed by the analysis of the filtered WoW traffic with our proposed method. The R/S plots also support the high correlation property of the time-series of the player states. The estimated H-parameter distribution for all the filtered and analyzed WoW traffic can be seen in Figure 7(c) with an average of $0.89$.

## V. Conclusion

In this paper, the traffic of the MMORPGs are examined with special focus on the effects of player behavior on macroscopic level of the gaming traffic. By understanding how the game traffic is influenced by the activity of the players and the changes in game environment, methods based on time series analysis are introduced to grab specific events and states during the game play.

The proposed methods were applied on traffic traces obtained from live operational broadband networks, in order to analyze the player behavior in real network situation. Both the user actions and the environment around the player were recognized in the MMORPG network traffic.

The collected data in the gaming environment was compared to data obtained from real world environment. We revealed the LRD property of the gaming traffic and also showed that the popular LRD explanation by heavy-tailed periods cannot be held. We also demonstrated that the player behavior affects the traffic characteristics on macroscopic level.

Our work can contribute to numerous research activities not restricting for the field of network science raising several additional questions and further work. The behavior of players can be compared to the behavior of people in real world environment and it is also possible to answer such questions that how well the virtual world models the real world.

## References


[1] S. Fernandes, C. Kamienski, , D. Sadok, J. Moreira, and R. Antonello, “Traffic analysis beyond this world: the case of second life,” in *NOSSDAV ’07*, Urbana, Illinois, USA, June 2007.

[2] K. Chen, P. Huang, C. Huang, and C. Lei, “Game traffic analysis: an MMORPG perspective,” in *NOSSDAV ’05*, New York, USA, 2005.

[3] “World of Warcraft,” http://www.worldofwarcraft.com.

[4] “Silkroad Online,” http://www.silkroadonline.net/.

[5] “Guild Wars,” http://www.guildwars.com/.

[6] “Eve Online,” http://www.eve-online.com/.

[7] “Star Wars Galaxies,” http://starwarsgalaxies.station.sony.com/.

[8] K. Chen, J. Jiang, P. Huang, H. Chu, C. Lei, and W. Chen, “Identifying MMORPG bots: A traffic analysis approach,” in *ACM SIGCHI ACE’06*, Los Angeles, USA, Jun 2006.

[9] A.-L. Barabási, “The origin of bursts and heavy tails in human dynamics,” *Nature*, vol. 435, p. 207, 2005.

[10] S. Mallat, “A theory for multiresolution signal decomposition: the wavelet representation,” *IEEE Pattern Anal. and Machine Intell.*, vol. 11, no. 7, pp. 674–693, 1989.

[11] “Matlab,” http://www.mathworks.com/.

[12] M. E. Crovella and A. Bestavros, “Self-similarity in world wide web traffic: Evidence and possible causes,” *IEEE/ACM Transactions on Networking*, vol. 5, pp. 835–846, 1997.

[13] G. Szabó, I. Szabó, and D. Orincsay, “Accurate traffic classification,” in *Proc. IEEE WOWMoM*, Helsinki, Finnland, June 2007.

[14] “The Daedalus Project,” http://www.nickyee.com/daedalus/archives/001586.php.

[15] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson, “Self-similarity through high-variability: Statistical analysis of ethernet lan traffic at the source level,” *IEEE/ACM Transactions on Networking 5*, vol. 5, pp. 71–86, 1997.

[16] A. Veres, Z. Kenesi, S. Molnár, and G. Vattay, “On the propagation of long-range dependence in the internet,” in *ACM SIGCOMM*, Stockholm, Sweden, USA, Aug 2000.

[17] J. Beran, “Statistics for long-memory processes,” *Chapman & Hall*, pp. 71–86, 1994.


## Appendix

```
function out = MMORPG_gamestate(rate)
nx   = length(ina);
out = zeros(size(ina));
[C5,L5]=wavedec(ina,5,'db1');A5 = wrcoef('a',C5,L5,'db1',5);
[C1,L1]=wavedec(ina,1,'db1');A1 = wrcoef('d',C1,L1,'db1',1);
medF=mean(abs(A1));maxF=max(A1);
phase_length=30; %depends on the wrcoef paratmeteres
state=0;
%init=stalling, just where?
if (abs(A5(1)-min(A5)) < abs(A5(1)-max(A5)))
   state=4;
else
   state=2;
end
for i=2:nx,
    frek_min=abs(abs(A1(i))-0);
    frek_max=abs(abs(A1(i))-medF);
    p=polyfit([1:phase_length*3],
              ina(i-phase_length:i+phase_length),1);
    %stalled outside, started to move
    if (state==4) && (frek_max<frek_min)
        state=3;
    elseif (state==2) && (frek_max<frek_min)
        state=1;
    %stalled and started
    elseif (state==3) && (frek_min<frek_max)
        state=4;
    elseif (state==1) && (frek_min<frek_max)
        state=2;
    end
    %it moved out to the desert
    if (state==1) && (A5(i)<A5(i-1) && p<0.5)
        state=3;
    elseif (state==3) && (A5(i)>A5(i-1) && p>0.5)
        state=1;
    end
    out(i)=state;
end % for i
```