SDDSR: Sequence Driven Dynamic Source Routing for Ad hoc Mobile Networks

by

Wael Kdouh

Submitted in Partial Fulfillment of the Requirements for the Degree of Masters of Science in Computer Science at the Lebanese American University

Thesis Advisor: Dr. Ramzi Haraty

Division of Computer Science and Mathematics
LEBANESE AMERICAN UNIVERSITY
August 2005

LEBANESE AMERICAN UNIVERSITY

Sequence Driven Dynamic Source Routing for Ad hoc Mobile Networks

Wael Kdouh

August 2005

LEBANESE AMERICAN UNIVERSITY

GRADUATE STUDIES

We hereby approve the thesis of

Wael Kdouh

Candidate for the Master of Science degree*.

[redacted]

Dr. Ramzi Haraty (Advisor)
Chairman, Division of Computer Science and Mathematics at the Lebanese American University

[redacted]

Dr. May Hamdan
Assistant Professor of Computer Science at the Lebanese American University

[redacted]

Dr. Faisal Abu Khzam
Assistant Professor of Computer Science at the Lebanese American University

* We also certify that written approval has been obtained for any proprietary material contained therein.

I grant to the LEBANESE AMERICAN UNIVERSITY the right to use this work, irrespective of any copyright, for the University's own purpose without cost to the University or to its students, agents and employees. I further agree that the University may reproduce and provide single copies of the work, in any format other than in or from microforms, to the public for the cost of reproduction.

_____

SDDSR: Sequence Driven Dynamic Source Routing for Ad hoc Mobile Networks

Abstract

By

Wael Kdouh

Mobile ad-hoc networks are becoming more popular as the use of mobile computers is increasing. The biggest challenge faced in such networks is continuous and random change in the topology. Table driven routing protocols where not designed for such networks. For this reason new routing protocols that can handle continuous change of topology where created. Two existing popular protocols for ad hoc mobile networks are Dynamic Source Routing (DSR), and Ad-hoc On-demand Distance Vector (AODV). DSR has the advantage of making heavy use of routing information to reduce the routing load, where as AODV has the advantage of using sequence numbers, which guarantees that at all time we are using non stale routing entries.

In this thesis we present the implementation of Sequence Driven Dynamic Source Routing (SDDSR). SDDSR is an on-demand routing protocol, which builds upon two existing on-demand routing protocols: DSR and AODV.

We use the NS-2 simulator to show the experimental results of the new protocol. The results showed better packet delivery as well as less routing load.

Keywords: Ad-hoc, Sequence Driven Dynamic Source Routing, Dynamic Source Routing, Ad-hoc On-demand Distance Vector, Network Simulator

Acknowledgements

I would like to thank Dr. Ramzi Haraty for his encouragement and support all the way. He has been and will always be a source of motivation for me. Simply it was an honor to be his student.

I would also like to thank my parents to whom I owe the greatest gratitude. I also wish to thank Mr. Omar Salloum, who taught me to perceive life from a different aspect, so a big thanks to him. Finally I would like to thank all the people who contributed in aiding me to accomplish this work and especially my source of inspiration, Lama Hussein.

Table of Contents

List of Figures

List of Tables

Chapter 1
Introduction

Wireless networks have been gaining popularity in recent years for many reasons [21]. The first reason is the innovative approach concerning the design of the wireless processors. Nowadays, we can see them integrated within compact electronic equipments like Pocket PCs. Another reason is the availability of those chips at cheap prices. An example is Intel's CMOS system, which is currently being manufactured and supports 802.11a, b and g, as well as the next generation 802.11n wireless networks, and hence this is an indication that wireless networks will keep on gaining even more popularity in the future. Another promising technology is Intels WIMAX, a technology which will boost the popular VOIP. This popularity of wireless technology yields to that of wireless mobile ad-hoc networks.

A mobile ad-hoc network is a peer-to-peer wireless network where there is no centralized access point which regulates the flow of data between hosts; rather each host participating in the network will act as a router.

In such networks, hosts usually move randomly from each other's range. Knowing this we can deduce that a lot of processing will have to be handled by each host. It is also obvious that there is no static graph of the overall network; rather this graph is being updated continuously.

As a result, an appropriate routing protocol is needed due to the limited resources of such hosts as well as the random nature of such networks which may end up increasing routing load.

Traditional table-driven routing protocols such as RIP, DSDV, CGSR, and WRP lack the capability to cope with the random change of topologies in mobile ad-hoc networks as shown in table 1 [7]. The aforementioned problem results from the fact that such protocols waste limited resources to discover routes that may never be needed. On the other hand, on-demand routing protocols have been introduced as solution of this problem [18]. Such protocols only attempt to initiate route discovery only when there is a request of communication between two hosts. Thus, less processing will be handled by each host since less route discoveries will be initiated.

Table 1. Overall comparison of on-demand versus table driven protocols.

Parameters	**On-demand**	**Table–Driven**
Availability of routing information	Available when needed	Always available regardless of need
Routing philosophy	Flat	Mostly flat
Periodic route updates	Not required	Required
Coping with mobility	Use localized route discovery	Inform other nodes to achieve a consistent routing table
Signaling traffic generated	Grows with increasing mobility	Greater than that of on-demand routing

There are many existing on-demand routing protocols such as AODV, DSR, TORA, ABR, SSR, and WRP [20]. The best existing on-demand routing protocols are DSR and AODV. DSR uses source routing and new routes are discovered only when needed. In other words, route discovery is only initiated when route request occurs, or route break down takes place. Although DSR surpasses table driven protocols, when high mobility occurs routing load tends to increase due to frequent link failures.

AODV, on the other hand, discovers routes in a similar procedure but without the use of source routing. AODV maintains tables instead of caching routes. A feature introduced in AODV is the use of sequence numbers to represent the freshness of routing information. Thus, AODV succeeds in delivering more packets and at the same time reduces the routing load. But AODV has its own problem, which is the number of route discovery request since it does not make full use of routing information. As a result the overall overhead increases.

Looking at the two protocols we can see that the advantage of DSR is the heavy use of routing information stored in each node's cache. Where as the advantage of AODV lies in the fact that it uses sequence numbers to avoid stale information and thus increase delivery ratio.

The new proposed protocol, Sequence Driven Dynamic Source Routing, uses the advantages of the two protocols. In other words it uses source routing combined with sequence numbers to enhance the overall performance.

The thesis is organized as follows. Chapter 2 defines the specifications and goals of the new protocol. Chapter 3 explores existing routing protocols for ad-hoc mobile

networks. Chapter 4 addresses the design and implementation of SDDSR. Chapter 5 reports the experimental results. Chapter 6 presents the conclusion.

Chapter 2
Literature Review

Wireless technology is becoming increasingly popular. It has been said that wireless networks will possibly become more widely used than wired networks. With the heavy use of personal digital assistants (PDAs), access to such networks has become a constant need.

Since the early 1970s, numerous protocols have been developed for ad hoc mobile networks. Such protocols dealt with the typical limitations of these networks, which include high power consumption, and low bandwidth [18]. In such networks a lot of overhead is introduced. Maltz and Johnson [10] stated that, this is caused by the presence of topological changes as well as broken routes. In such networks, the high mobility, low bandwidth, and limited computing capability characteristics of mobile hosts make the design of routing protocols challenging. The protocols must be able to keep up with the drastically and unpredictably changing network topology, with minimized message exchanges.

The routing protocols may be categorized as table driven, or on-demand, according to the way the mobile hosts exchange routing information. The table driven protocols, such as DSDV [1] and CGSR [1], periodically disseminate routing information among all the hosts in the network, so that every host has the up-to-date information for all possible routes. On-demand routing protocols, such as Ad-hoc On-demand Distance Vector [3] and Dynamic Source Routing [5], operate on a need basis, discover and

maintain only active routes that are currently used for delivering data packets. On demand protocols behavior leads to performance gain in terms of both, routing load as well as well as packet delivery [28].

J. Raju and J. J. Garcia-Luna-Acceves [28] argued that table-driven routing could be as efficient as on-demand routing, by relaxing the optimal or shortest path requirement of the former. But even with such enhancements, on-demand routing protocols still proved to show less routing overhead [13] . So for this reason, we find nowadays that on-demand protocols are being used as the routing protocols for ad-hoc networks.

Since the two most representative on-demand routing protocols are AODV [3,8,9,11,19] and DSR [4,5,10,24,25], several researchers have tried to improve those protocols. Perkins [3] edited the original version of AODV and Johnson [5] edited the original version of DSR, in order to cope with the nature of ad-hoc networks.

In spite of the enhancements done by Johnson and Perkins, the overall mechanism was still the same. For this reason we decided to come up with a new protocol that builds upon the two existing protocols, but with some enhancements. The power of DSR lies under its use of source routing as stated by Johnson, where by the power of AODV lies under its use of sequence numbers as stated by Perkins.

In this thesis, we present a new protocol that uses source routing as well as sequence numbers in order to obtain better results concerning both, routing load as well as packet delivery. In the next sections we discuss the two existing protocols in details as well as the implementation of the new protocol.

Chapter 3
Specifications and Goals

Like all on-demand routing protocols, Sequence Driven Dynamic Source Routing (SDDSR) consists of three phases: (a) route discovery, (b) route setup, (c) route maintenance.

What are the objectives behind the development and the implementation of SDDSR? SDDSR aims at to improve packet delivery ratio, as well as reducing routing load. Packet delivery ratio can be improved by the use of sequence numbers, which guarantees at all times the use of fresh routes. As for the routing load, it is solved through two stages. The first stage is the heavy use of information stored in each node's cache, which results in less route requests. Where as the second stage ensures that the amount of information carried during route discovery is always minimized.

1. SDDSR Phases

Like any on-demand routing protocol SDDSR is going to include three phases:

1.1 Route Discovery

Route discovery allows any host in the ad-hoc network to dynamically discover a route to any other host in the ad hoc network, whether directly reachable within wireless transmission range or reachable through one or more intermediate network hops through other hosts. A host initiating a route discovery broadcasts a route request packet, which may be received by those hosts within

wireless transmission range of it. The route request packet identifies the host for which the route is requested. If the route discovery is successful, the initiating host receives a route reply packet listing a sequence of hops through which the target is reachable. The route request propagates through the network until it reaches the destination or an intermediate node possessing a route to the destination. This node is responsible for sending the route reply.

1.2 Route Setup

This phase is also known as route reply, it involves sending back a reply to the source of the request. This reply will eventually inform the source about the route to destination. The replying host could be the destination itself or any intermediate host depending on the validity of the entry as will be shown in chapter 4.

1.3 Route Maintenance:

This phase is responsible for identifying any broken links in the network that may be caused by a host going out of transmission range, or by a sudden halt of a host. Post to failure identification, a route request is initiated triggering the whole three phases to take place again.

2. SDDSR Performance Analysis

The new protocol is evaluated based on two key metrics:

(i) Packet Delivery Ratio--- the ratio of the packets delivered to the destination to those generated by the CBR (Constant Bit Rate) sources.

(ii) Routing Load---the number of routing packets "transmitted" per data packet "delivered" at the destination.

Other important parameters to evaluate the protocol performance are shown in table 2.

Table 2. Performance parameters of three on-demand routing protocols.

Performance parameters	**AODV**	**DSR**	**SDDSR**
Time complexity (initialization)	O(2d)	O(2d)	O(2d)
Time complexity (postfailure)	O(2d)	O(2d)	O(2d)
Communication complexity (postfailure)	O(2n)	O(2n)	O(2n)
Loop-free	Yes	Yes	Yes
Routes maintained in	Route table	Route cache	Route cache
Route reconfiguration methodology	Erase route; notify source	Erase route; notify source	Erase route; notify source
Routing metric	Freshest and shortest path	Shortest path	Freshest and shortest path

*d= Diameter of the network
*n= Number of nodes

Since SDDSR is based on both DSR and AODV, it is expected to show similar results as those shown in table 2 [14]. For this reason, the performance evaluation is going to be based on the two key performance parameters:

(i) Packet delivery ratio, and

(ii) Routing load

There is a very note

Chapter 4
Existing On-Demand Routing Protocols for Ad-hoc Mobile Networks

This chapter introduces two existing on-demand routing protocols which have proved to be the leading protocols pertaining to Ad-hoc networks: **DSR** and **AODV**. These two protocols are going to form the base of **SDDSR**.

1. Ad-hoc On-demand Distance Vector (AODV) Routing Protocol

AODV is an on-demand routing protocol as its name implies. That is, it only initiates a discovery request when needed. Each node also includes a table to track all received requests [3,8,9,11,19]. A request is usually identified by the combination of a broadcast ID and source address. If a request has been processed previously, it will be ignored in order to prevent the network from being flooded with the same request more than once and at the same time ensure that the shortest path is always achieved.

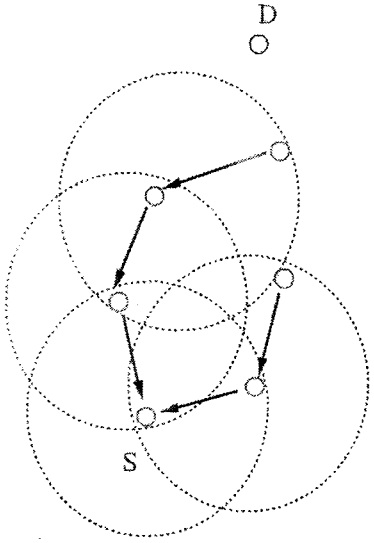
AODV uses traditional routing tables, and at any time there is only one entry for each destination. In order to achieve this it always gives the priority to the routes with the highest sequence number. In case two routes have the same sequence numbers, the one with smaller number of hops is chosen. It uses sequence numbers to ensure the freshness of routing information and to prevent loops. These sequence numbers are carried by all routing packets [13]. Each node has a table indicating the latest sequence number of all other nodes.

AODV utilizes a timer for each entry in the table in order to track entries that are not valid anymore. A routing table entry is expired after a certain time, which is usually determined according to the network itself. A set of predecessor nodes is maintained in each routing table entry, which denotes the set of neighboring nodes that use this entry to route packets. Those nodes are notified with route error (RERR) packets when link failure occurs. The process of RERR propagation is done in a recursive manner, where each node notifies its predecessors until it reaches the source of the RREQ.

When the RREQ reaches the destination or an intermediate node which has a valid entry in its routing table, the reverse path to the source will be ready as shown in figure 1.

When the source node gets a reply from the destination or an intermediate node, it will be able to start sending data to that node only. If it needs to send a packet to any other node, it has to start a RREQ all over again. This is a disadvantage since it does not make heavy use of routing information in each route request and as a result it will increase the overall routing load. This is a problem that is going to be dealt with in the proposed protocol.

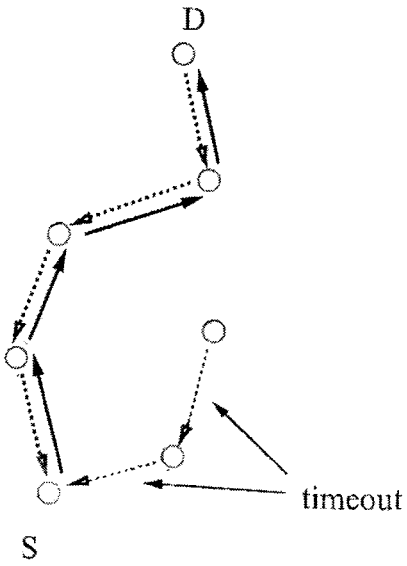
Figure 2 shows the forward path formation. It also shows how the route entries not lying on the request path will time out after a period of time. The timeout period is usually dynamic depending on the network itself.



*S=Source

*D=Destination

Figure 1. Reverse path formation.



*S=Source

*D=Destination

Figure 2. Forward path formation.

2. Dynamic Source Routing (DSR) Protocol

Unlike AODV, DSR uses source routing. That is, the sender knows the complete hop route to the destination as shown in figure 3.

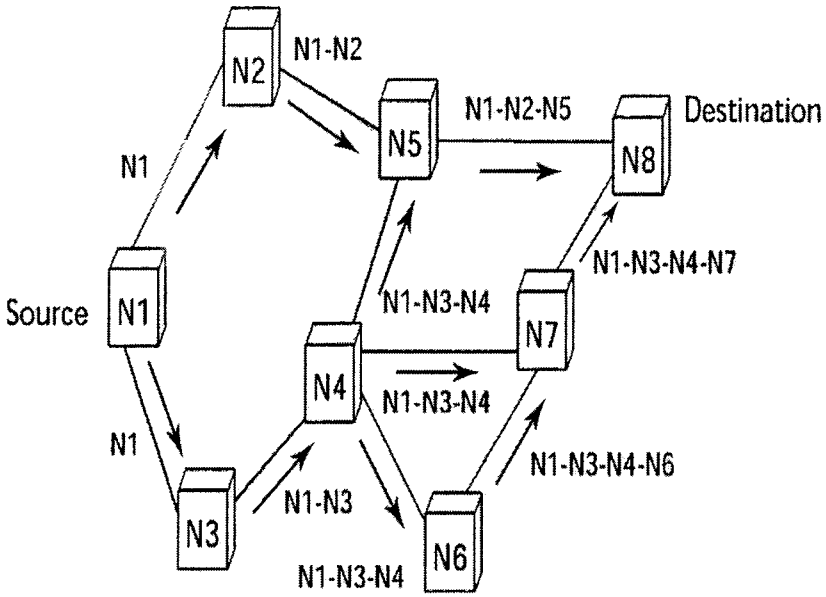


Figure 3. Formation of route record during route request [1].

This is where the power of DSR lies since it helps reduce the number of route requests and thus reduce the routing load. But this is not always the case, because when a lot of link failures occur, the overall performance tends to decrease [4].

When a node needs to send data packet to a destination for which it does not know the route, it uses a route discovery process to dynamically determine such a route. Like any on–demand routing protocol it starts by sending a RREQ. Each node receiving a RREQ rebroadcasts it, unless it is the destination itself or it possesses route in its cache to the destination [24,25].

Each node includes a table to track all received requests. A request is usually identified by the combination of a broadcast ID and source address. If a request has been processed previously, it will be ignored in order to prevent the network from being flooded with route requests.

In case the RREQ reaches the destination itself, the source route included within the RREQ itself will be reversed and used by the RREP to reach the source of the route request as shown in figure 4 [5]. Incase the node containing a route to the destination is an intermediate node, it appends the route in its cache after making sure that it is valid, or in other words it is unexpired, to the route traversed so far. As for the loops they can never exist because no request can be processed more than once by the same node as mentioned before.

Beside the problem of frequent linkage failures which causes routing load, another major problem is usually faced when using DSR which is the usage of stale routes [10].

Since DSR does not utilize any sequence number as AODV does, and has no mechanism to prevent more than one route to the same destination, there is no way to ensure that at all time the latest route is used, and hence this leads to less packet delivery ratio. This is a problem that is going to be dealt with in the proposed protocol.

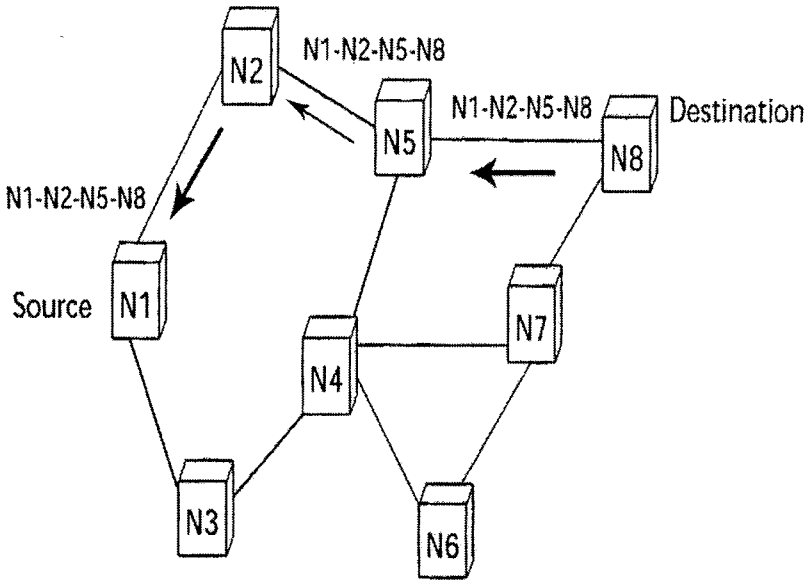


Figure 4. Propagation of route reply.

3. Performance Comparison of DSR and AODV

Although the two protocols share the on-demand nature, each has its advantages and disadvantages. DSR has the advantage of making heavy use of the routing information, while AODV has the advantage of using up to date routing information, which results in higher packet delivery ratio. These facts are reflected in figure 5 and figure 6.

The results in figure 5 shows that the packet delivery ratio tends to improve under AODV as the number of sources increase and the pause time decreases. This is because the more sources we have and the faster the movement of the nodes become, the more link failures are to occur. As for the results in figure 6, they show clear difference in performance between DSR and AODV. This is the result of the heavy use of routing information under DSR.

Looking at the results of both figure 5 as well as Figure 6, we can see that the combination of AODV's packet delivery ratio performance with DSR's routing load performance, would lead to an improved on-demand routing protocol. This fact is going to form the base of the proposed protocol.

Although both have their advantages and disadvantages, DSR tends have a better overall performance [16], so for this reason the new protocol results are going to be compared with DSR.

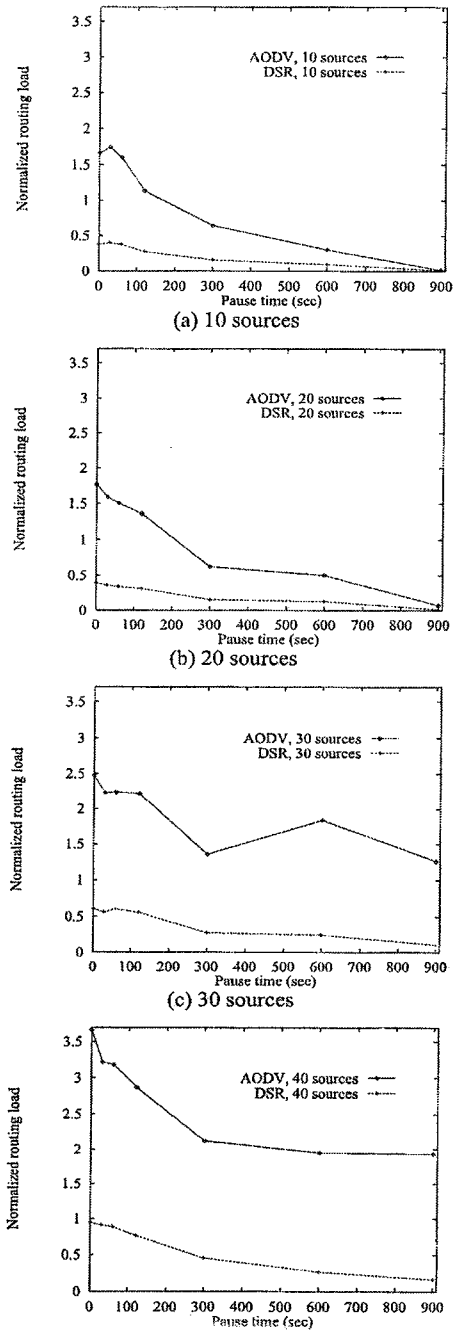


Figure 5. Normalized routing load for 50 node model with various number of sources [13].

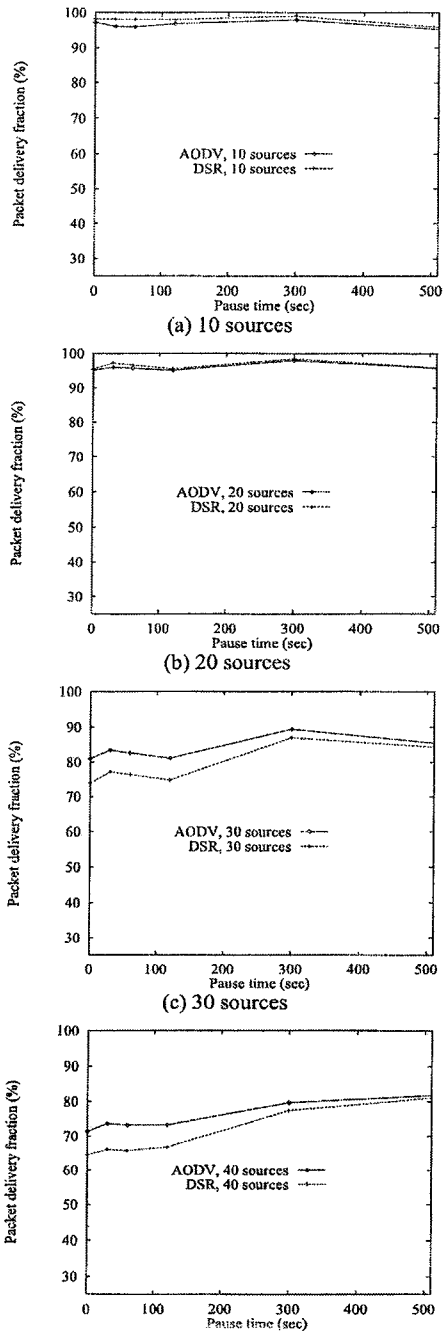


Figure 6. Packet delivery ratio for 50 node model with various number of sources[13].

Chapter 5
SDDSR

This chapter presents the implementation of SDDSR. As mentioned previously it is based on two existing on-demand routing protocols: AODV and DSR.

Destination Sequenced Dynamic Source Routing an on-demand routing protocol. Like most of the protocols of its category, it consists of three phases:

i) Route discovery,

ii) Route set-up, and

iii) Route maintenance.

1. The Algorithm

Every host **i** has a request number ($\mathbf{rn_i}$), a sequence number ($\mathbf{sn_i}$), and two tables for storing the last known request number for each host **j** ($\mathbf{rn_i[j]}$) and the last known sequence number for each destination ($\mathbf{sn_i[j]}$). The request number $\mathbf{rn_i}$ along with address will be used by the receiving nodes to identify the request of node **i**. As for the sequence number $\mathbf{sn_i}$, it will be used to make sure that only fresh routes to node **i**, are going to be used, and hence improve the delivery ratio.

SDDSR uses caching in stead of routing tables, since it is going to utilize source routing in the route setup. To overcome the problem of DSR [10,12,17,25], where the addresses of all nodes had to be carried during both, the route discovery as well as the route reply, SDDSR does not use source routing in discovering a route to the destination. In stead it uses the same mechanism as AODV, where every node along the path will point to the previous node from which it received the request. This helps in reducing the routing load. So by the time the request will reach the destination, or any other node acquiring a valid route to the destination, it will know the way back to the source. After the route to the destination is discovered, a reply containing the addresses of all nodes (source routing) has to be built during the route reply. As the reply propagates back towards the source of the request, each node will add its address as well as its latest sequence number. The addresses will eventually be used to build the source route where as the sequence numbers will be used to update the tables of all nodes on the way back to make that all time they contain the latest sequence numbers of all the nodes along the active path. The use of sequence numbers also ensures that the network is loop free. However, the use of cached routes imposes some changes on the way sequence numbers are used. The use of cached routes is important to make use of all routing information gathered in the route discovery phase, which will lead us to solve the problem of AODV, since AODV uses sequence numbers without exploiting all routing information.

In the following subsections, we explain the implementation of the three phases, which constitute SDDSR.

1.1. Route Discovery

Since SDDSR is an on-demand protocol, it implies that a host initiates a route discovery only when it needs a route to the destination. Suppose there is a host **s** that needs to send a data packet to a destination host **d**. First, host **s** checks whether it has a route to the destination, and in case it does not, it will increase its counter $\mathbf{rn_s}$, before initiating a route request which has the following structure:

$<\mathbf{s, pn, rn_s, sn_{maxd}, d}>$ where **pn** denotes the address of the host from which the packet was received (originally **pn=s**), $\mathbf{rn_s}$ the request number of the source host, and $\mathbf{sn_{maxd}}$ the maximum sequence number recorded during the packet's travel (initially $\mathbf{sn_{maxd}= sn_s[d]}$, where $\mathbf{sn_s[d]}$ is the sequence number of host **d** known at host **s**). Each host **i** that receives the RREQ, checks the numbers $< \mathbf{s, rn_s} >$. If $\mathbf{rn_s} < \mathbf{rn_i}[s]$ then the RREQ is discarded, otherwise $\mathbf{rn_i}[s]$ gets updated, and host **i** performs a series of actions: i) if number $\mathbf{sn_i}[d] > \mathbf{sn_{maxd}}$, then the packet is updated with the value $\mathbf{sn_i}[d]$, ii) adds in its cache a route entry that contains the data $< \mathbf{pn, s, rn_s} >$, which will be used later as a reverse route to reach the host from which the packet was received.

This route entry is used in setting up the discovered route, iii) the packet is finally broadcasted. After a period of time, the RREQ packet will reach either the destination or an intermediate node possessing a "valid" route to the destination. In either case, if the receiving host has not processed the RREQ before, it increases its sequence number and starts the route setup phase.

1.2. Route setup

After the RREQ packet reaches the destination or an intermediate node possessing a "valid" route to the destination, the route setup is started by that node. It involves replying to the requesting node by sending back the discovered route. Let us consider the address of the replying host is **r**, and $<\mathbf{x}_{r,j,d}>$ the vector containing the addresses of the hosts consisting the j-th route to **d** existing in the route cache of host **r**, and $<\mathbf{sn}_{r,j,d}>$ the vector of the corresponding sequence numbers. The reply packet includes the following vectors:

$$<\mathbf{x}_{r,j,d}> = <\mathbf{r},\mathbf{i}_1, \mathbf{i}_2,\ldots, \mathbf{i}_{k-1},\mathbf{d}> \quad (1)$$

$$<\mathbf{sn}_{r,j,}> = <\mathbf{sn}_r, \mathbf{sn}_{rj}\,[\mathbf{i}_1],\ldots, \mathbf{sn}_{rj}\,[\mathbf{i}_{k-1}], \mathbf{sn}_{rj}\,[\mathbf{d}]> \quad (2)$$

$\mathbf{sn}_{rj}$ **[i]** is the sequence number of host **i** at the time **j** route was formed. It is clear that in the case that r=d the aforementioned vectors are simplified to [**d**] and $[\mathbf{sn}_d]$.

Host **pn** will receive a reply packet from **r** that contains the following information$<\mathbf{s},\mathbf{rn}_s,\mathbf{x}_{r,j,d},\mathbf{sn}_{r,j,d}>$. After receiving the RREP packet, **pn** caches the new route and updates the numbers $\mathbf{sn}_{pn}$**[i]**, for any **i** belonging to $<\mathbf{x}_{r,j,d}>$. In addition it updates its sequence number $\mathbf{sn}_{pn}$, before adding it along with its

address to the RREP packet. Finally, using the numbers <**s**, $\mathbf{rn_s}$ >, it recalls the next hop to host **s** in order to transmit the packet. The reverse path pointing towards the source node **s** is deleted after a period of $\mathbf{t_{exp}}$ seconds. This is done to allow multiple replies to reach the originator of the request.

A new feature of SDDSR is the control of cache reply. Contrary to DSR not all of hosts having a route to a destination can reply to a request for this destination. If for example a host **r** receives a packet originated from host **s**, requesting a route to host **d**, even if host **r** has a route in its cache to the destination, it can not use it before making sure that this route is valid. A route is only considered valid when:

$$\mathbf{sn_r[d] > sn_{maxd}} \qquad (3)$$

where $\mathbf{sn_{maxd}}$ is the sequence number recorder in the packet header. In this way not only more up to data routes are used but it also the formation of loops is excluded.

1.3. Route Maintenance and Packet Forwarding

SDDSR uses source routing in forwarding data packets. As for broken links it identifies them in the same way as DSR. That is, if a host tries to send a packet for several times and does not get any reply, it will send RERR to the source of the request. When the host receives the RERR, it will remove all the routes containing the hosts forming the broken link, and a new route discovery process will be initiated by the source, if the route is still in need.

Chapter 6
Experimental Results

1. The Simulator

SDDSR was evaluated under NS 2 (Network Simulator version 2). NS is built to run under UNIX, so we used CYGWIN, which emulates UNIX platform under windows in order to compile it [27]. NS is written in C++ and OTcl (Tcl script language with Object-oriented extensions) [15]. NS is an event-driven network simulator that simulates variety of IP networks. It implements network protocols such as TCP and UPD, traffic source behavior such as FTP, Telnet, Web and CBR, router queue management mechanism such as Drop Tail and RED, routing algorithms such as Dijkstra.

As shown in figure 7, NS is an object-oriented Tcl (OTcl) script interpreter that has a simulation event scheduler and network component object libraries, and network setup (plumbing) module libraries (actually, plumbing modules are implemented as member functions of the base simulator object).

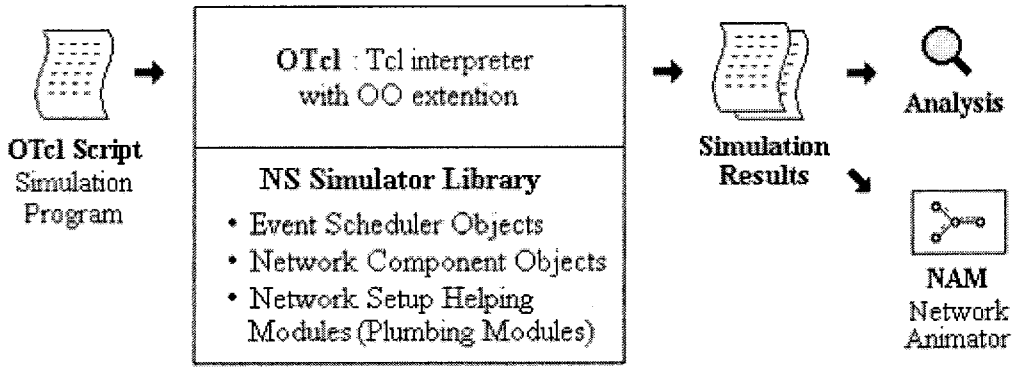


Figure 7. Structure of NS [27].

To setup and run a simulation network, an OTcl scripts must be written to initiate an event scheduler, setup the network topology using the network objects and the plumbing functions in the library, and tell the traffic sources when to start and stop transmitting packets through the event scheduler.

The term "plumbing" is used for a network setup, because setting up a network is plumbing possible data paths among network objects by setting the "neighbor" pointer of an object to the address of an appropriate object. The power of NS comes from this plumbing.

MobileNode is the basic node object with added functionalities like movement, ability to transmit and receive on a channel that allows it to be used to create mobile, wireless simulation environments. The class MobileNode is derived from the base class Node. The mobility features including node movement, periodic position updates, maintaining topology boundary, etc. are implemented in C++ while plumbing of network components within MobileNode itself (like classifiers, dmux, LL, Mac, Channel, etc.) have been implemented in Otcl.

2. Simulation Results

In order to asses the performance of SDDSR and DSR, we created the code under C++ as well as OTcl. The simulated scenario consisted of ten nodes moving randomly within a boundary of 500x500m^2. The simulation runs for 400 *seconds*. We used the Random Waypoint Mobility to control the node movement movement as well as transmission range [2,22].

After running the simulation, the results showed that SDDSR outperforms DSR in terms of both, routing load as well as packet delivery ratio.

The script below was written under OTcl in order to assess the performance of the new protocol compared to that of DSR.

```
# Define options
=============================================================
set val(chan)           Channel/WirelessChannel
set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(x)    500     ;# X dimension of the topography
set val(y)    500     ;# Y dimension of the topography
set val(ifqlen)            50      ;# max packet in ifq
set val(seed)              0.0
set val(adhocRouting)      DSR
set val(nn)    11      ;# how many nodes are simulated
set val(movement)       "tcl/mobility/scene/my-ovement"
set val(traffic)        "tcl/mobility/scene/my-traffic"
set val(stop) 400       ;# simulation time
```

```
#=====================================================
# Main Program
#=====================================================

# create simulator instance

set ns_  [new Simulator]

# setup topography object

set topo  [new Topography]

# create trace object for ns

set tracefd     [open thesis.tr w]

$ns_ trace-all $tracefd


# define topology
$topo load_flatgrid $val(x) $val(y)

# Create God
set god_ [create-god $val(nn)]

# define how node should be created

#global node setting

$ns_ node-config -adhocRouting $val(adhocRouting) \
                 -llType $val(ll) \
                 -macType $val(mac) \
                 -ifqType $val(ifq) \
                 -ifqLen $val(ifqlen) \
                 -antType $val(ant) \
                 -propType $val(prop) \
                 -phyType $val(netif) \
                 -channelType $val(chan) \
                  -topoInstance $topo \
                  -agentTrace ON \
                 -routerTrace OFF \
                 -macTrace OFF
```

```
#  Create the specified number of nodes [$val(nn)]
#  and "attach" them to the channel.

for {set i 0} {$i < $val(nn) } {incr i} {
     set node_($i) [$ns_ node]
     $node_($i) random-motion 0 ;# disable random
                                # motion
}

# Define node movement model

puts "Loading connection pattern..."
source $val(movement)

# Define traffic model

puts "Loading scenario file..."
source $val(traffic)


# Tell nodes when the simulation ends

for {set i 0} {$i < $val(nn) } {incr i} {
    $ns_ at $val(stop).0 "$node_($i) reset";
}

$ns_ at   $val(stop).0002  "puts  \"NS  EXITING...\"  ;  $ns_
halt"


puts "Starting Simulation..."
$ns_ run
```

The script was run twice, once using DSR, and the other using SDDSR. The above was run using DSR, as for the one below it was run using SDDSR.

```
# Define options
=========================================================
set val(chan)           Channel/WirelessChannel
set val(prop)           Propagation/TwoRayGround
set val(netif)          Phy/WirelessPhy
set val(mac)            Mac/802_11
set val(ifq)            Queue/DropTail/PriQueue
set val(ll)             LL
set val(ant)            Antenna/OmniAntenna
set val(x)    500     ;# X dimension of the topography
set val(y)    500     ;# Y dimension of the topography
set val(ifqlen)            50      ;# max packet in ifq
set val(seed)              0.0
set val(adhocRouting)   SDDSR
set val(nn)    11      ;# how many nodes are simulated
set val(movement)    "tcl/mobility/scene/my-ovement"
set val(traffic)     "tcl/mobility/scene/my-traffic"
set val(stop) 400    ;# simulation time

=====================================================
# Main Program
=====================================================
# create simulator instance

set ns_   [new Simulator]

# setup topography object

set topo  [new Topography]

# create trace object for ns

set tracefd     [open thesis.tr w]

$ns_ trace-all $tracefd

# define topology
$topo load_flatgrid $val(x) $val(y)

# Create God
set god_ [create-god $val(nn)]

# define how node should be created
```

```
#global node setting

$ns_ node-config -adhocRouting $val(adhocRouting) \
                 -llType $val(ll) \
                 -macType $val(mac) \
                 -ifqType $val(ifq) \
                 -ifqLen $val(ifqlen) \
                 -antType $val(ant) \
                 -propType $val(prop) \
                 -phyType $val(netif) \
                 -channelType $val(chan) \
                  -topoInstance $topo \
                  -agentTrace ON \
                 -routerTrace OFF \
                 -macTrace OFF


#  Create the specified number of nodes [$val(nn)] #  and
"attach" them to the channel.

for {set i 0} {$i < $val(nn) } {incr i} {
     set node_($i) [$ns_ node]
     $node_($i) random-motion 0 ;# disable random
                                # motion
}

# Define node movement model

puts "Loading connection pattern..."
source $val(movement)


# Define traffic model

puts "Loading scenario file..."
source $val(traffic)


# Tell nodes when the simulation ends

for {set i 0} {$i < $val(nn) } {incr i} {
    $ns_ at $val(stop).0 "$node_($i) reset";
}

$ns_ at   $val(stop).0002  "puts  \"NS  EXITING...\"  ;  $ns_
halt"
```

```
puts "Starting Simulation..."
$ns_ run
```

Figure 8 below shows that the packet delivery ratio of SDDSR is much higher than that of the DSR protocol. The reason for the high packet delivery ratio for SDDSR is that it never uses stale routes. If it doesn't have a route for a destination it finds one through "Route Discovery" and it also uses "Route Maintenance" to ensure that only valid routes are stored at the nodes. The graph represents the result of 10 nodes moving at different speeds. As shown in the graph, the more mobility we tend to have in the network, the better SDDSR tend to perform. We can see that at the speed of 20 m/sec SDDSR performs 87% delivery ratio where as DSR performs 78%.

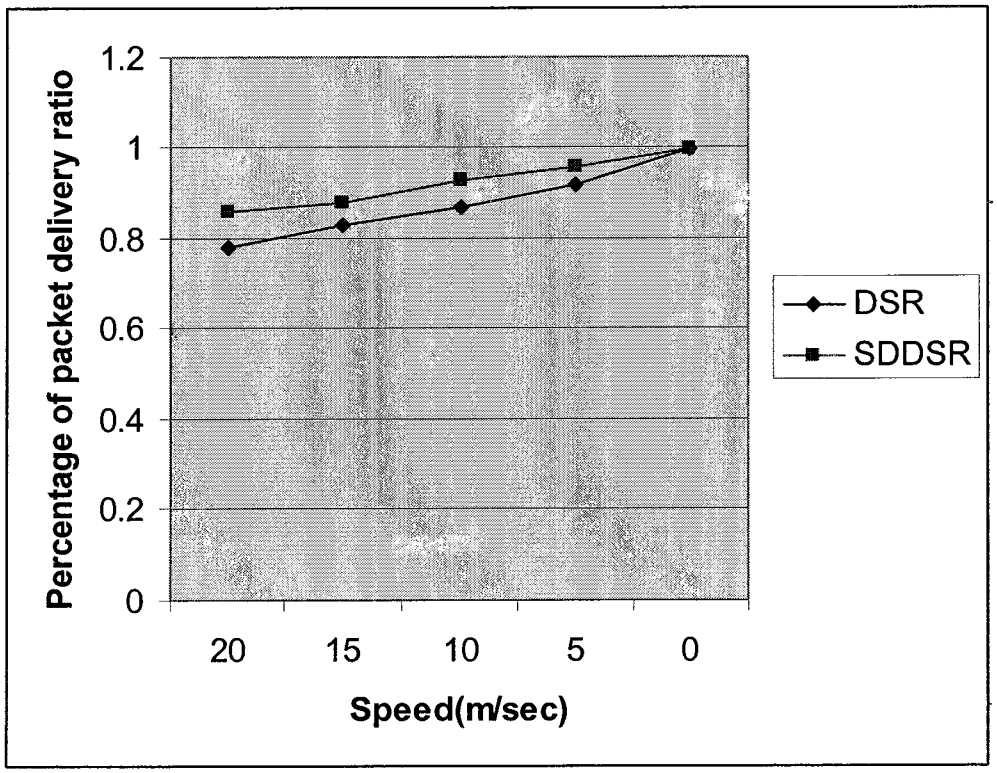


Figure 8: Delivery ratio for different speeds.

Figure 9 below shows that the performance of SDDSR is again better. The better results under SDDSR are due to the use of fresh routing information. The routing overhead is the highest for low pause times. As the movement of the nodes becomes less frequent the routes persist for larger durations of time and hence the number of "Route Discovery" packets decreases. This explains the decrease in the routing overhead as the pause time increases.

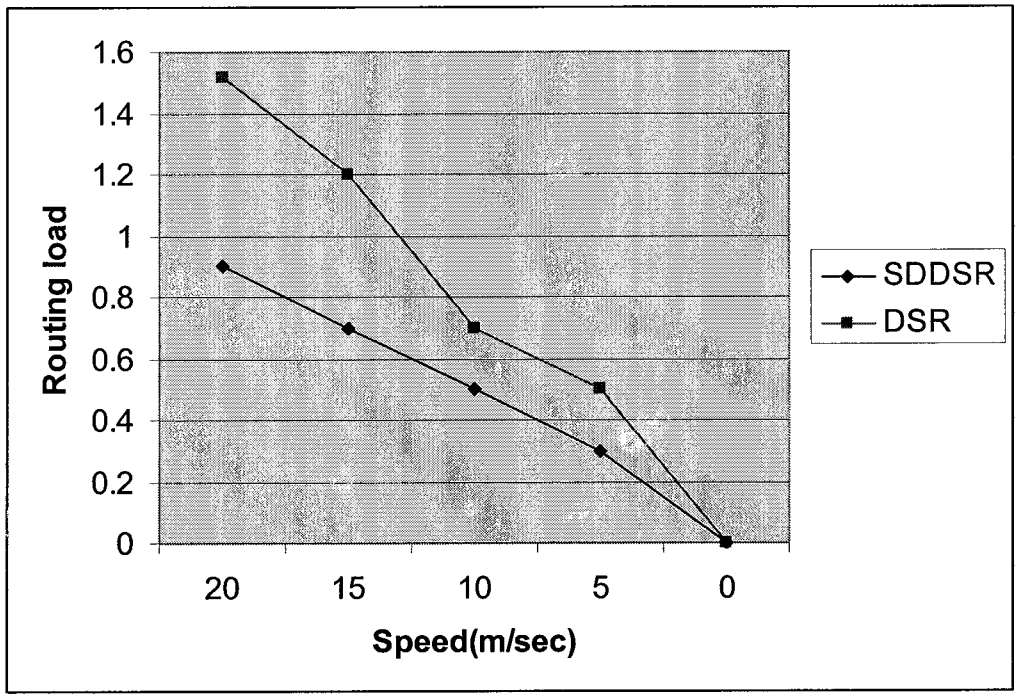


Figure 9: Routing load for different speeds.

Chapter 7
Conclusion and Further Work

In this thesis we developed a new routing protocol for mobile ad-hoc networks. The new protocol uses source routing, caching, and sequence numbers, to decrease routing load and improve delivery ratio. The results showed that SDDSR outperforms DSR in terms of both delivery ratio and routing load. Moreover, SDDSR does not use source routing during route discovery in order to avoid the waste of useful network resources, a problem which existed in DSR and caused the increase the routing load.

We plan to continue with our simulations for DSR and SDDSR. We would like to study the effects of changing other parameters e.g. the number of mobile nodes, the dimensions of simulation space and the speed of movement of the nodes and collect other metrics if possible. We can also repeat each test with a large number of scenarios to remove the randomness if any from the results.

SDDSR can be enhanced by using the Q-routing algorithm [26], which will further improve the performance of SDDSR concerning routing load and packet delivery ratio. Q-routing can further reduce the overall routing load by replacing shortest path mechanism with reinforcement learning module.

Although Q-routing algorithm is expected to improve the overall performance, its implementation should under go some changes in order to cope with the on going change of topology that occurs under ad-hoc networks.

Appendix A: Results Explanation

Before interpreting the produced results, let us explain the meaning of each field. The first field can have one of three values: **s**, **r**, or **d.** Where **s** stands for send, **r** for receive and **d** for dropped. The second field shows the number of the node receiving or sending the packet. The third field is always going to be **AGT,** because as shown in the script, only agent tracing was turned on. Although router trace is turned off in the script, we may still see **RTR** in the third field. This will only occur when a packets gets dropped.

The rest of the fields are informatory and are not necessary to interpret the results.

1. SDDSR Results:

The following are the results of the script using SDDSR as the routing protocol.

```
s 2.556838879 _1_ AGT  --- 0 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [0] 0 1
r 2.569378597 _2_ AGT  --- 0 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [0] 1 1
s 2.681923773 _1_ AGT  --- 1 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [1] 0 1
r 2.687652662 _2_ AGT  --- 1 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [1] 1 1
s 2.908303542 _1_ AGT  --- 2 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [2] 0 1
r 2.914272431 _2_ AGT  --- 2 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [2] 1 1
s 3.080144853 _1_ AGT  --- 3 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [3] 0 1
r 3.086173741 _2_ AGT  --- 3 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [3] 1 1
s 3.427981808 _1_ AGT  --- 4 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [4] 0 1
r 3.433770697 _2_ AGT  --- 4 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [4] 1 1
s 3.555683893 _1_ AGT  --- 5 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [5] 0 1
r 3.561312781 _2_ AGT  --- 5 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [5] 1 1
s 3.711459767 _1_ AGT  --- 6 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [6] 0 1
r 3.716908655 _2_ AGT  --- 6 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [6] 1 1
s 4.011128073 _1_ AGT  --- 7 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [7] 0 1
r 4.016736962 _2_ AGT  --- 7 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [7] 1 1
s 4.201584110 _1_ AGT  --- 8 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [8] 0 1
r 4.207332998 _2_ AGT  --- 8 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [8] 1 1
s 4.333473478 _1_ AGT  --- 9 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [9] 0 1
```

```
r 4.339222366 _2_ AGT  --- 9 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [9] 1 1
s 4.654720254 _1_ AGT  --- 10 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [10] 0 1
r 4.660649142 _2_ AGT  --- 10 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [10] 1 1
s 4.831403761 _1_ AGT  --- 11 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [11] 0 1
r 4.837152649 _2_ AGT  --- 11 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [11] 1 1
s 5.200405436 _1_ AGT  --- 12 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [12] 0 1
r 5.205974324 _2_ AGT  --- 12 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [12] 1 1
s 5.337938787 _1_ AGT  --- 13 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [13] 0 1
r 5.343467675 _2_ AGT  --- 13 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [13] 1 1
s 5.646711569 _1_ AGT  --- 14 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [14] 0 1
r 5.652220457 _2_ AGT  --- 14 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [14] 1 1
s 5.975311607 _1_ AGT  --- 15 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [15] 0 1
r 5.980880495 _2_ AGT  --- 15 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [15] 1 1
s 6.318973183 _1_ AGT  --- 16 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [16] 0 1
r 6.324542071 _2_ AGT  --- 16 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [16] 1 1
s 6.644476149 _1_ AGT  --- 17 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [17] 0 1
r 6.650465037 _2_ AGT  --- 17 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [17] 1 1
s 6.973371734 _1_ AGT  --- 18 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [18] 0 1
r 6.979120622 _2_ AGT  --- 18 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [18] 1 1
s 7.135868842 _1_ AGT  --- 19 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [19] 0 1
r 7.141897730 _2_ AGT  --- 19 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [19] 1 1
s 7.354269803 _1_ AGT  --- 20 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [20] 0 1
r 7.359878691 _2_ AGT  --- 20 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [20] 1 1
s 7.562323410 _1_ AGT  --- 21 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [21] 0 1
r 7.567892298 _2_ AGT  --- 21 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [21] 1 1
s 7.695521499 _1_ AGT  --- 22 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [22] 0 1
r 7.700970387 _2_ AGT  --- 22 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [22] 1 1
s 7.703020315 _7_ AGT  --- 23 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [0] 0 2
r 7.737220551 _9_ AGT  --- 23 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [0] 2 2
s 7.821359066 _1_ AGT  --- 24 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [23] 0 1
r 7.827327954 _2_ AGT  --- 24 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [23] 1 1
s 8.048019224 _7_ AGT  --- 25 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [1] 0 2
r 8.059803597 _9_ AGT  --- 25 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [1] 2 2
s 8.144868559 _1_ AGT  --- 26 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [24] 0 1
r 8.150817447 _2_ AGT  --- 26 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [24] 1 1
s 8.230263442 _7_ AGT  --- 27 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [2] 0 2
r 8.241907814 _9_ AGT  --- 27 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [2] 2 2
s 8.509298441 _1_ AGT  --- 28 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [25] 0 1
r 8.514887329 _2_ AGT  --- 28 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [25] 1 1
s 8.591519460 _7_ AGT  --- 29 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [3] 0 2
r 8.603203830 _9_ AGT  --- 29 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [3] 2 2
s 8.763805904 _1_ AGT  --- 30 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [26] 0 1
r 8.769814792 _2_ AGT  --- 30 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [26] 1 1
s 8.782234680 _7_ AGT  --- 31 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [4] 0 2
r 8.794199049 _9_ AGT  --- 31 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [4] 2 2
s 9.001894008 _7_ AGT  --- 32 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [5] 0 2
```

r 9.013738375 _9_ AGT --- 32 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [5] 2 2
s 9.053120025 _1_ AGT --- 33 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [27] 0 1
r 9.058948913 _2_ AGT --- 33 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [27] 1 1
s 9.168506378 _7_ AGT --- 34 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [6] 0 2
r 9.180190744 _9_ AGT --- 34 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [6] 2 2
s 9.191599045 _1_ AGT --- 35 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [28] 0 1
r 9.197127933 _2_ AGT --- 35 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [28] 1 1
s 9.307343034 _7_ AGT --- 36 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [7] 0 2
r 9.319007399 _9_ AGT --- 36 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [7] 2 2
s 9.464803416 _7_ AGT --- 37 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [8] 0 2
r 9.476527780 _9_ AGT --- 37 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [8] 2 2
s 9.524110395 _1_ AGT --- 38 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [29] 0 1
r 9.530079283 _2_ AGT --- 38 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [29] 1 1
s 9.804146596 _7_ AGT --- 39 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [9] 0 2
r 9.815610958 _9_ AGT --- 39 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [9] 2 2
s 9.858684327 _1_ AGT --- 40 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [30] 0 1
r 9.864473215 _2_ AGT --- 40 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [30] 1 1
s 9.996585471 _7_ AGT --- 41 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [10] 0 2
r 10.008589831 _9_ AGT --- 41 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [10] 2 2
s 10.060671479 _1_ AGT --- 42 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [31] 0 1
r 10.066540367 _2_ AGT --- 42 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [31] 1 1
s 10.187525139 _7_ AGT --- 43 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [11] 0 2
s 10.195625367 _1_ AGT --- 44 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [32] 0 1
r 10.199189498 _9_ AGT --- 43 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [11] 2 2
r 10.205647516 _2_ AGT --- 44 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [32] 1 1
s 10.409443717 _1_ AGT --- 45 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [33] 0 1
r 10.415052605 _2_ AGT --- 45 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [33] 1 1
s 10.438801476 _7_ AGT --- 46 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [12] 0 2
r 10.450605833 _9_ AGT --- 46 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [12] 2 2
s 10.571868562 _1_ AGT --- 47 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [34] 0 1
r 10.577537450 _2_ AGT --- 47 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [34] 1 1
s 10.596402995 _7_ AGT --- 48 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [13] 0 2
r 10.608247351 _9_ AGT --- 48 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [13] 2 2
s 10.790676659 _1_ AGT --- 49 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [35] 0 1
r 10.796405547 _2_ AGT --- 49 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [35] 1 1
s 10.828375851 _7_ AGT --- 50 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [14] 0 2
r 10.840540206 _9_ AGT --- 50 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [14] 2 2
s 11.045701994 _1_ AGT --- 51 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [36] 0 1
r 11.051690881 _2_ AGT --- 51 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [36] 1 1
s 11.184324909 _7_ AGT --- 52 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [15] 0 2
r 11.196149261 _9_ AGT --- 52 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [15] 2 2
s 11.288958531 _1_ AGT --- 53 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [37] 0 1
r 11.294407419 _2_ AGT --- 53 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [37] 1 1
s 11.371519585 _7_ AGT --- 54 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [16] 0 2
r 11.382823936 _9_ AGT --- 54 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [16] 2 2
s 11.541698799 _1_ AGT --- 55 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [38] 0 1

```
r 11.547287687 _2_ AGT  --- 55 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [38] 1 1
s 11.689549846 _7_ AGT  --- 56 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [17] 0 2
r 11.700934195 _9_ AGT  --- 56 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [17] 2 2
s 11.813640985 _1_ AGT  --- 57 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [39] 0 1
r 11.819469873 _2_ AGT  --- 57 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [39] 1 1
s 11.877372349 _7_ AGT  --- 58 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [18] 0 2
r 11.889176696 _9_ AGT  --- 58 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [18] 2 2
s 12.102067501 _7_ AGT  --- 59 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [19] 0 2
r 12.113631847 _9_ AGT  --- 59 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [19] 2 2
s 12.147603365 _1_ AGT  --- 60 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [40] 0 1
r 12.153552253 _2_ AGT  --- 60 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [40] 1 1
s 12.302935526 _7_ AGT  --- 61 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [20] 0 2
r 12.314799870 _9_ AGT  --- 61 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [20] 2 2
s 12.337041776 _1_ AGT  --- 62 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [41] 0 1
r 12.342990664 _2_ AGT  --- 62 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [41] 1 1
s 12.449197705 _7_ AGT  --- 63 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [21] 0 2
r 12.461382049 _9_ AGT  --- 63 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [21] 2 2
s 12.473903897 _1_ AGT  --- 64 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [42] 0 1
r 12.479632785 _2_ AGT  --- 64 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [42] 1 1
s 12.649804560 _1_ AGT  --- 65 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [43] 0 1
r 12.655773448 _2_ AGT  --- 65 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [43] 1 1
s 12.804863392 _7_ AGT  --- 66 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [22] 0 2
r 12.816507733 _9_ AGT  --- 66 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [22] 2 2
s 12.962891634 _7_ AGT  --- 67 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [23] 0 2
s 12.974097199 _1_ AGT  --- 68 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [44] 0 1
r 12.974935974 _9_ AGT  --- 67 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [23] 2 2
r 12.981393988 _2_ AGT  --- 68 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [44] 1 1
s 13.109250568 _7_ AGT  --- 69 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [24] 0 2
r 13.120694907 _9_ AGT  --- 69 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [24] 2 2
s 13.169363964 _1_ AGT  --- 70 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [45] 0 1
r 13.174972852 _2_ AGT  --- 70 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [45] 1 1
s 13.333043181 _1_ AGT  --- 71 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [46] 0 1
r 13.338532069 _2_ AGT  --- 71 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [46] 1 1
s 13.439203256 _7_ AGT  --- 72 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [25] 0 2
r 13.451107594 _9_ AGT  --- 72 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [25] 2 2
s 13.526336765 _1_ AGT  --- 73 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [47] 0 1
r 13.531785654 _2_ AGT  --- 73 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [47] 1 1
s 13.767690058 _1_ AGT  --- 74 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [48] 0 1
r 13.773178947 _2_ AGT  --- 74 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [48] 1 1
s 13.773581192 _7_ AGT  --- 75 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [26] 0 2
r 13.785572345 _9_ AGT  --- 75 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [26] 2 2
s 13.959838501 _1_ AGT  --- 76 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [49] 0 1
r 13.965367389 _2_ AGT  --- 76 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [49] 1 1
s 13.991506695 _7_ AGT  --- 77 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [27] 0 2
r 14.003111028 _9_ AGT  --- 77 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [27] 2 2
s 14.132656593 _1_ AGT  --- 78 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [50] 0 1
```

r 14.138185481 _2_ AGT --- 78 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [50] 1 1
s 14.260048848 _7_ AGT --- 79 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [28] 0 2
r 14.272113179 _9_ AGT --- 79 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [28] 2 2
s 14.444465431 _1_ AGT --- 80 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [51] 0 1
r 14.450374319 _2_ AGT --- 80 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [51] 1 1
s 14.471609931 _7_ AGT --- 81 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [29] 0 2
r 14.483434262 _9_ AGT --- 81 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [29] 2 2
s 14.669005033 _7_ AGT --- 82 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [30] 0 2
r 14.681009361 _9_ AGT --- 82 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [30] 2 2
s 14.697826120 _1_ AGT --- 83 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [52] 0 1
r 14.703675009 _2_ AGT --- 83 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [52] 1 1
s 14.886814777 _7_ AGT --- 84 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [31] 0 2
r 14.898039104 _9_ AGT --- 84 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [31] 2 2
s 14.922110750 _1_ AGT --- 85 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [53] 0 1
r 14.927899638 _2_ AGT --- 85 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [53] 1 1
s 15.091304205 _1_ AGT --- 86 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [54] 0 1
r 15.096993094 _2_ AGT --- 86 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [54] 1 1
s 15.196026853 _7_ AGT --- 87 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [32] 0 2
r 15.207431178 _9_ AGT --- 87 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [32] 2 2
s 15.360007114 _7_ AGT --- 88 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [33] 0 2
r 15.371671439 _9_ AGT --- 88 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [33] 2 2
s 15.418620996 _1_ AGT --- 89 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [55] 0 1
r 15.424229884 _2_ AGT --- 89 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [55] 1 1
s 15.589889037 _1_ AGT --- 90 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [56] 0 1
r 15.595537925 _2_ AGT --- 90 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [56] 1 1
s 15.728314389 _7_ AGT --- 91 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [34] 0 2
r 15.740038711 _9_ AGT --- 91 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [34] 2 2
s 15.904567608 _1_ AGT --- 92 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [57] 0 1
r 15.910476497 _2_ AGT --- 92 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [57] 1 1
s 16.036901903 _1_ AGT --- 93 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [58] 0 1
r 16.042550792 _2_ AGT --- 93 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [58] 1 1
s 16.048537078 _7_ AGT --- 94 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [35] 0 2
r 16.060261397 _9_ AGT --- 94 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [35] 2 2
s 16.184819494 _1_ AGT --- 95 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [59] 0 1
r 16.190548383 _2_ AGT --- 95 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [59] 1 1
s 16.407781762 _7_ AGT --- 96 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [36] 0 2
r 16.419366079 _9_ AGT --- 96 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [36] 2 2
s 16.537170902 _1_ AGT --- 97 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [60] 0 1
r 16.542719791 _2_ AGT --- 97 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [60] 1 1
s 16.613924709 _7_ AGT --- 98 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [37] 0 2
r 16.625769025 _9_ AGT --- 98 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [37] 2 2
s 16.752275214 _7_ AGT --- 99 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [38] 0 2
r 16.764279529 _9_ AGT --- 99 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [38] 2 2
s 16.882055194 _1_ AGT --- 100 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [61] 0 1
r 16.887644082 _2_ AGT --- 100 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [61] 1 1
s 17.073878541 _7_ AGT --- 101 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [39] 0 2

```
r 17.085642854 _9_ AGT  --- 101 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [39] 2 2
s 17.133163725 _1_ AGT  --- 102 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [62] 0 1
r 17.138652614 _2_ AGT  --- 102 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [62] 1 1
s 17.328742976 _7_ AGT  --- 103 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [40] 0 2
r 17.340367287 _9_ AGT  --- 103 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [40] 2 2
s 17.464668967 _1_ AGT  --- 104 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [63] 0 1
r 17.470357856 _2_ AGT  --- 104 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [63] 1 1
s 17.651091272 _7_ AGT  --- 105 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [41] 0 2
r 17.662555581 _9_ AGT  --- 105 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [41] 2 2
s 17.699884082 _1_ AGT  --- 106 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [64] 0 1
r 17.705672971 _2_ AGT  --- 106 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [64] 1 1
s 17.994686644 _7_ AGT  --- 107 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [42] 0 2
r 18.006530951 _9_ AGT  --- 107 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [42] 2 2
s 18.018222690 _1_ AGT  --- 108 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [65] 0 1
r 18.023871580 _2_ AGT  --- 108 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [65] 1 1
s 18.237451059 _7_ AGT  --- 109 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [43] 0 2
r 18.248675364 _9_ AGT  --- 109 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [43] 2 2
s 18.363693748 _1_ AGT  --- 110 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [66] 0 1
r 18.369442638 _2_ AGT  --- 110 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [66] 1 1
s 18.567321600 _7_ AGT  --- 111 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [44] 0 2
r 18.579205903 _9_ AGT  --- 111 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [44] 2 2
s 18.631779587 _1_ AGT  --- 112 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [67] 0 1
r 18.637368477 _2_ AGT  --- 112 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [67] 1 1
s 18.711584885 _7_ AGT  --- 113 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [45] 0 2
r 18.722889188 _9_ AGT  --- 113 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [45] 2 2
s 18.962137817 _1_ AGT  --- 114 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [68] 0 1
r 18.967766706 _2_ AGT  --- 114 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [68] 1 1
s 18.971776575 _7_ AGT  --- 115 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [46] 0 2
r 18.983900875 _9_ AGT  --- 115 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [46] 2 2
s 19.270483710 _1_ AGT  --- 116 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [69] 0 1
r 19.276532600 _2_ AGT  --- 116 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [69] 1 1
s 19.294399648 _7_ AGT  --- 117 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [47] 0 2
r 19.306343946 _9_ AGT  --- 117 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [47] 2 2
s 19.466532240 _1_ AGT  --- 118 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [70] 0 1
r 19.472361130 _2_ AGT  --- 118 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [70] 1 1
s 19.476449002 _7_ AGT  --- 119 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [48] 0 2
r 19.488593300 _9_ AGT  --- 119 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [48] 2 2
s 19.639573016 _1_ AGT  --- 120 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [71] 0 1
r 19.645241906 _2_ AGT  --- 120 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [71] 1 1
s 19.816224899 _7_ AGT  --- 121 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [49] 0 2
s 19.825102481 _1_ AGT  --- 122 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [72] 0 1
r 19.827909194 _9_ AGT  --- 121 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [49] 2 2
r 19.834287200 _2_ AGT  --- 122 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [72] 1 1
s 20.005374752 _1_ AGT  --- 123 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [73] 0 1
r 20.010983642 _2_ AGT  --- 123 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [73] 1 1
s 20.109831097 _7_ AGT  --- 124 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [50] 0 2
```

r 20.121535390 _9_ AGT --- 124 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [50] 2 2
s 20.187462980 _1_ AGT --- 125 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [74] 0 1
r 20.193491870 _2_ AGT --- 125 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [74] 1 1
s 20.237994405 _7_ AGT --- 126 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [51] 0 2
r 20.249918697 _9_ AGT --- 126 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [51] 2 2
s 20.485484684 _8_ AGT --- 127 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [0] 0 2
s 20.501493451 _1_ AGT --- 128 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [75] 0 1
r 20.513376126 _2_ AGT --- 128 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [75] 1 1
r 20.523625970 _9_ AGT --- 127 cbr 532 [13a 9 4 800] ------- [8:1 9:1 28 9] [0] 3 2
s 20.564611844 _7_ AGT --- 129 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [52] 0 2
r 20.576596134 _9_ AGT --- 129 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [52] 2 2
s 20.717073975 _1_ AGT --- 130 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [76] 0 1
r 20.723082865 _2_ AGT --- 130 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [76] 1 1
s 20.790319098 _8_ AGT --- 131 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [1] 0 2
r 20.801903194 _9_ AGT --- 131 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [1] 2 2
s 20.884351782 _7_ AGT --- 132 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [53] 0 2
r 20.895796069 _9_ AGT --- 132 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [53] 2 2
s 20.934892724 _1_ AGT --- 133 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [77] 0 1
r 20.940481614 _2_ AGT --- 133 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [77] 1 1
s 21.073162421 _7_ AGT --- 134 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [54] 0 2
r 21.084986708 _9_ AGT --- 134 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [54] 2 2
s 21.105825667 _1_ AGT --- 135 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [78] 0 1
r 21.111394557 _2_ AGT --- 135 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [78] 1 1
s 21.149980604 _8_ AGT --- 136 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [2] 0 2
r 21.161764697 _9_ AGT --- 136 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [2] 2 2
s 21.289122006 _1_ AGT --- 137 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [79] 0 1
r 21.294850896 _2_ AGT --- 137 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [79] 1 1
s 21.346900288 _8_ AGT --- 138 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [3] 0 2
s 21.358146817 _7_ AGT --- 139 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [55] 0 2
r 21.359204379 _9_ AGT --- 138 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [3] 2 2
r 21.371838080 _9_ AGT --- 139 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [55] 2 2
s 21.604915695 _7_ AGT --- 140 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [56] 0 2
s 21.616296702 _8_ AGT --- 141 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [4] 0 2
r 21.616619978 _9_ AGT --- 140 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [56] 2 2
r 21.629493382 _9_ AGT --- 141 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [4] 2 2
s 21.647625685 _1_ AGT --- 142 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [80] 0 1
r 21.653514576 _2_ AGT --- 142 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [80] 1 1
s 21.804764923 _8_ AGT --- 143 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [5] 0 2
r 21.816329010 _9_ AGT --- 143 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [5] 2 2
s 21.963181198 _1_ AGT --- 144 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [81] 0 1
s 21.967518768 _7_ AGT --- 145 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [57] 0 2
r 21.968970089 _2_ AGT --- 144 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [81] 1 1
r 21.981223401 _9_ AGT --- 145 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [57] 2 2
s 22.159995021 _8_ AGT --- 146 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [6] 0 2
r 22.171659106 _9_ AGT --- 146 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [6] 2 2
s 22.244646143 _1_ AGT --- 147 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [82] 0 1

r 22.250295033 _2_ AGT --- 147 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [82] 1 1
s 22.303773122 _7_ AGT --- 148 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [58] 0 2
r 22.315637400 _9_ AGT --- 148 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [58] 2 2
s 22.379138997 _8_ AGT --- 149 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [7] 0 2
r 22.390843080 _9_ AGT --- 149 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [7] 2 2
s 22.427499153 _1_ AGT --- 150 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [83] 0 1
r 22.433328044 _2_ AGT --- 150 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [83] 1 1
s 22.492281628 _7_ AGT --- 151 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [59] 0 2
r 22.504405906 _9_ AGT --- 151 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [59] 2 2
s 22.506308908 _8_ AGT --- 152 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [8] 0 2
r 22.518232991 _9_ AGT --- 152 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [8] 2 2
s 22.703521198 _7_ AGT --- 153 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [60] 0 2
r 22.715045474 _9_ AGT --- 153 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [60] 2 2
s 22.771653692 _1_ AGT --- 154 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [84] 0 1
r 22.777262583 _2_ AGT --- 154 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [84] 1 1
s 22.821408494 _8_ AGT --- 155 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [9] 0 2
r 22.833232573 _9_ AGT --- 155 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [9] 2 2
s 22.994241565 _1_ AGT --- 156 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [85] 0 1
r 23.000230456 _2_ AGT --- 156 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [85] 1 1
s 23.069490606 _7_ AGT --- 157 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [61] 0 2
r 23.081694879 _9_ AGT --- 157 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [61] 2 2
s 23.151418077 _8_ AGT --- 158 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [10] 0 2
r 23.163382155 _9_ AGT --- 158 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [10] 2 2
s 23.357511146 _1_ AGT --- 159 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [86] 0 1
r 23.363120037 _2_ AGT --- 159 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [86] 1 1
s 23.427683992 _7_ AGT --- 160 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [62] 0 2
r 23.439508264 _9_ AGT --- 160 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [62] 2 2
s 23.448403883 _8_ AGT --- 161 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [11] 0 2
r 23.459987958 _9_ AGT --- 161 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [11] 2 2
s 23.674146666 _8_ AGT --- 162 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [12] 0 2
r 23.685970739 _9_ AGT --- 162 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [12] 2 2
s 23.711953643 _1_ AGT --- 163 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [87] 0 1
r 23.717862535 _2_ AGT --- 163 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [87] 1 1
s 23.800799008 _7_ AGT --- 164 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [63] 0 2
r 23.812743277 _9_ AGT --- 164 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [63] 2 2
s 23.872133995 _8_ AGT --- 165 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [13] 0 2
r 23.883878067 _9_ AGT --- 165 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [13] 2 2
s 23.974060026 _1_ AGT --- 166 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [88] 0 1
r 23.979748918 _2_ AGT --- 166 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [88] 1 1
s 23.997909193 _8_ AGT --- 167 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [14] 0 2
r 24.009593264 _9_ AGT --- 167 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [14] 2 2
s 24.117203435 _7_ AGT --- 168 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [64] 0 2
r 24.129407702 _9_ AGT --- 168 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [64] 2 2
s 24.158196700 _8_ AGT --- 169 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [15] 0 2
r 24.170020770 _9_ AGT --- 169 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [15] 2 2
s 24.237950637 _1_ AGT --- 170 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [89] 0 1

r 24.243959529 _2_ AGT --- 170 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [89] 1 1
s 24.357779024 _8_ AGT --- 171 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [16] 0 2
r 24.369403092 _9_ AGT --- 171 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [16] 2 2
s 24.407217462 _7_ AGT --- 172 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [65] 0 2
r 24.418881727 _9_ AGT --- 172 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [65] 2 2
s 24.558349964 _8_ AGT --- 173 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [17] 0 2
r 24.570194030 _9_ AGT --- 173 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [17] 2 2
s 24.574828224 _1_ AGT --- 174 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [90] 0 1
r 24.580817115 _2_ AGT --- 174 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [90] 1 1
s 24.686008751 _7_ AGT --- 175 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [66] 0 2
r 24.697573015 _9_ AGT --- 175 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [66] 2 2
s 24.734352793 _1_ AGT --- 176 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [91] 0 1
r 24.740301685 _2_ AGT --- 176 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [91] 1 1
s 24.800340055 _8_ AGT --- 177 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [18] 0 2
r 24.812144119 _9_ AGT --- 177 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [18] 2 2
s 24.834121834 _7_ AGT --- 178 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [67] 0 2
r 24.845726096 _9_ AGT --- 178 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [67] 2 2
s 25.011960256 _7_ AGT --- 179 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [68] 0 2
s 25.021435889 _8_ AGT --- 180 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [19] 0 2
r 25.023724518 _9_ AGT --- 179 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [68] 2 2
r 25.035917887 _9_ AGT --- 180 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [19] 2 2
s 25.073838515 _1_ AGT --- 181 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [92] 0 1
r 25.079427408 _2_ AGT --- 181 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [92] 1 1
s 25.175328019 _7_ AGT --- 182 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [69] 0 2
r 25.187572279 _9_ AGT --- 182 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [69] 2 2
s 25.194025857 _8_ AGT --- 183 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [20] 0 2
r 25.206049918 _9_ AGT --- 183 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [20] 2 2
s 25.328752812 _7_ AGT --- 184 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [70] 0 2
r 25.340237071 _9_ AGT --- 184 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [70] 2 2
s 25.402578324 _8_ AGT --- 185 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [21] 0 2
r 25.414642384 _9_ AGT --- 185 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [21] 2 2
s 25.436672687 _1_ AGT --- 186 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [93] 0 1
r 25.442321580 _2_ AGT --- 186 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [93] 1 1
s 25.456380409 _7_ AGT --- 187 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [71] 0 2
r 25.468364667 _9_ AGT --- 187 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [71] 2 2
s 25.637388722 _1_ AGT --- 188 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [94] 0 1
r 25.643397615 _2_ AGT --- 188 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [94] 1 1
s 25.692848759 _8_ AGT --- 189 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [22] 0 2
r 25.704472816 _9_ AGT --- 189 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [22] 2 2
s 25.748835572 _7_ AGT --- 190 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [72] 0 2
r 25.760299829 _9_ AGT --- 190 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [72] 2 2
s 25.818640205 _1_ AGT --- 191 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [95] 0 1
r 25.824329097 _2_ AGT --- 191 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [95] 1 1
s 25.904631179 _8_ AGT --- 192 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [23] 0 2
r 25.915995234 _9_ AGT --- 192 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [23] 2 2
s 25.975877807 _7_ AGT --- 193 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [73] 0 2

```
r 25.987962062 _9_ AGT  --- 193 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [73] 2 2
s 26.035245856 _8_ AGT  --- 194 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [24] 0 2
r 26.046929911 _9_ AGT  --- 194 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [24] 2 2
s 26.067566999 _1_ AGT  --- 195 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [96] 0 1
r 26.073475892 _2_ AGT  --- 195 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [96] 1 1
s 26.208352145 _7_ AGT  --- 196 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [74] 0 2
r 26.219656398 _9_ AGT  --- 196 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [74] 2 2
s 26.246833645 _1_ AGT  --- 197 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [97] 0 1
r 26.252562538 _2_ AGT  --- 197 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [97] 1 1
s 26.396307880 _8_ AGT  --- 198 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [25] 0 2
r 26.408371931 _9_ AGT  --- 198 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [25] 2 2
s 26.452156312 _7_ AGT  --- 199 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [75] 0 2
r 26.463500564 _9_ AGT  --- 199 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [75] 2 2
s 26.513546814 _1_ AGT  --- 200 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [98] 0 1
r 26.519215707 _2_ AGT  --- 200 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [98] 1 1
s 26.543646956 _8_ AGT  --- 201 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [26] 0 2
r 26.555651006 _9_ AGT  --- 201 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [26] 2 2
s 26.673812980 _1_ AGT  --- 202 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [99] 0 1
r 26.679321873 _2_ AGT  --- 202 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [99] 1 1
s 26.681602664 _8_ AGT  --- 203 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [27] 0 2
s 26.686402640 _7_ AGT  --- 204 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [76] 0 2
r 26.693526714 _9_ AGT  --- 203 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [27] 2 2
r 26.706220366 _9_ AGT  --- 204 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [76] 2 2
s 26.831971907 _7_ AGT  --- 205 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [77] 0 2
r 26.844156157 _9_ AGT  --- 205 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [77] 2 2
s 27.038475773 _8_ AGT  --- 206 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [28] 0 2
s 27.048641169 _1_ AGT  --- 207 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [100] 0 1
r 27.050039819 _9_ AGT  --- 206 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [28] 2 2
r 27.056277818 _2_ AGT  --- 207 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [100] 1 1
s 27.125527280 _7_ AGT  --- 208 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [78] 0 2
r 27.137211527 _9_ AGT  --- 208 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [78] 2 2
s 27.230448739 _8_ AGT  --- 209 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [29] 0 2
s 27.240008974 _1_ AGT  --- 210 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [101] 0 1
r 27.242092784 _9_ AGT  --- 209 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [29] 2 2
r 27.248370783 _2_ AGT  --- 210 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [101] 1 1
s 27.330674432 _7_ AGT  --- 211 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [79] 0 2
r 27.342298678 _9_ AGT  --- 211 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [79] 2 2
s 27.406541668 _1_ AGT  --- 212 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [102] 0 1
r 27.412070561 _2_ AGT  --- 212 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [102] 1 1
s 27.473089903 _7_ AGT  --- 213 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [80] 0 2
r 27.484834148 _9_ AGT  --- 213 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [80] 2 2
s 27.533098599 _8_ AGT  --- 214 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [30] 0 2
r 27.544722641 _9_ AGT  --- 214 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [30] 2 2
s 27.666544314 _1_ AGT  --- 215 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [103] 0 1
r 27.672193207 _2_ AGT  --- 215 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [103] 1 1
s 27.731061343 _7_ AGT  --- 216 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [81] 0 2
```

```
r 27.742805587 _9_ AGT  --- 216 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [81] 2 2
s 27.803302687 _8_ AGT  --- 217 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [31] 0 2
r 27.815246728 _9_ AGT  --- 217 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [31] 2 2
s 27.856039966 _1_ AGT  --- 218 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [104] 0 1
r 27.861768860 _2_ AGT  --- 218 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [104] 1 1
s 28.021565275 _8_ AGT  --- 219 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [32] 0 2
r 28.033589314 _9_ AGT  --- 219 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [32] 2 2
s 28.061560505 _7_ AGT  --- 220 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [82] 0 2
r 28.073664747 _9_ AGT  --- 220 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [82] 2 2
s 28.210690113 _1_ AGT  --- 221 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [105] 0 1
r 28.216179006 _2_ AGT  --- 221 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [105] 1 1
s 28.267499268 _8_ AGT  --- 222 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [33] 0 2
r 28.279623305 _9_ AGT  --- 222 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [33] 2 2
s 28.300388466 _7_ AGT  --- 223 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [83] 0 2
r 28.312312706 _9_ AGT  --- 223 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [83] 2 2
s 28.465331764 _1_ AGT  --- 224 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [106] 0 1
s 28.468405891 _8_ AGT  --- 225 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [34] 0 2
r 28.470980658 _2_ AGT  --- 224 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [106] 1 1
r 28.483493537 _9_ AGT  --- 225 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [34] 2 2
s 28.603408533 _8_ AGT  --- 226 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [35] 0 2
s 28.612487970 _7_ AGT  --- 227 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [84] 0 2
r 28.615092568 _9_ AGT  --- 226 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [35] 2 2
r 28.627286204 _9_ AGT  --- 227 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [84] 2 2
s 28.732228894 _8_ AGT  --- 228 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [36] 0 2
r 28.744192927 _9_ AGT  --- 228 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [36] 2 2
s 28.774863225 _1_ AGT  --- 229 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [107] 0 1
r 28.780352119 _2_ AGT  --- 229 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [107] 1 1
s 28.959413189 _7_ AGT  --- 230 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [85] 0 2
r 28.971417426 _9_ AGT  --- 230 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [85] 2 2
s 29.036746525 _1_ AGT  --- 231 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [108] 0 1
r 29.042775419 _2_ AGT  --- 231 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [108] 1 1
s 29.105121996 _8_ AGT  --- 232 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [37] 0 2
r 29.116986027 _9_ AGT  --- 232 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [37] 2 2
s 29.152205537 _7_ AGT  --- 233 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [86] 0 2
r 29.164229772 _9_ AGT  --- 233 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [86] 2 2
s 29.214320108 _1_ AGT  --- 234 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [109] 0 1
r 29.220009003 _2_ AGT  --- 234 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [109] 1 1
s 29.284168952 _8_ AGT  --- 235 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [38] 0 2
r 29.296052981 _9_ AGT  --- 235 cbr 532 [13a 9 4 800] ------- [8:1 9:1 29 9] [38] 2 2
s 29.389738365 _7_ AGT  --- 236 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [87] 0 2
r 29.401822598 _9_ AGT  --- 236 cbr 532 [13a 9 4 800] ------- [7:2 9:0 29 9] [87] 2 2
s 29.546173154 _7_ AGT  --- 237 cbr 512 [0 0 0 0] ------- [7:1 8:0 32 0] [0] 0 1
r 29.555190288 _8_ AGT  --- 237 cbr 532 [13a 8 7 800] ------- [7:1 8:0 30 8] [0] 1 1
s 29.555813164 _1_ AGT  --- 238 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [110] 0 1
r 29.564419431 _2_ AGT  --- 238 cbr 532 [13a 2 1 800] ------- [1:0 2:0 30 2] [110] 1 1
```

D 29.568044957 _5_ RTR TTL 0 DSDSR 48 [0 ffffffff 3 800] ------- [5:255 -1:255 0 0] [0x2 4 2 [8 5] [7 6]] (REQUEST)

2. DSR Results:

The following are the results of the script using SDDSR as the routing protocol.

s 2.556838879 _1_ AGT --- 0 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [0] 0 1
SFESTs 2.570510197 _1_ 0 [1 -> 2] 1(1) to 2 [1 |2]
r 2.576671085 _2_ AGT --- 0 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [0] 1 1
s 2.821150271 _1_ AGT --- 3 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [1] 0 1
SFESTs 2.821150271 _1_ 3 [1 -> 2] 1(1) to 2 [1 |2]
r 2.827167160 _2_ AGT --- 3 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [1] 1 1
s 3.034348523 _1_ AGT --- 4 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [2] 0 1
SFESTs 3.034348523 _1_ 4 [1 -> 2] 1(1) to 2 [1 |2]
r 3.040485411 _2_ AGT --- 4 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [2] 1 1
s 3.371226109 _1_ AGT --- 5 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [3] 0 1
SFs 3.371226109 _1_ 5 [1 -> 2] 1(0) to 2
r 3.377194998 _2_ AGT --- 5 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [3] 1 1
s 3.679170887 _1_ AGT --- 6 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [4] 0 1
SFs 3.679170887 _1_ 6 [1 -> 2] 1(0) to 2
r 3.684859776 _2_ AGT --- 6 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [4] 1 1
s 3.906753445 _1_ AGT --- 7 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [5] 0 1
SFs 3.906753445 _1_ 7 [1 -> 2] 1(0) to 2
r 3.912802333 _2_ AGT --- 7 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [5] 1 1
s 4.107660068 _1_ AGT --- 8 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [6] 0 1
SFs 4.107660068 _1_ 8 [1 -> 2] 1(0) to 2
r 4.113428956 _2_ AGT --- 8 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [6] 1 1
s 4.418776862 _1_ AGT --- 9 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [7] 0 1
SFs 4.418776862 _1_ 9 [1 -> 2] 1(0) to 2
r 4.424765750 _2_ AGT --- 9 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [7] 1 1
s 4.772057699 _1_ AGT --- 10 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [8] 0 1
SFs 4.772057699 _1_ 10 [1 -> 2] 1(0) to 2
r 4.777826588 _2_ AGT --- 10 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [8] 1 1
s 5.009629612 _1_ AGT --- 11 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [9] 0 1
SFs 5.009629612 _1_ 11 [1 -> 2] 1(0) to 2
r 5.015278501 _2_ AGT --- 11 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [9] 1 1
s 5.256838407 _1_ AGT --- 12 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [10] 0 1
SFs 5.256838407 _1_ 12 [1 -> 2] 1(0) to 2
r 5.262887295 _2_ AGT --- 12 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [10] 1 1
s 5.534904583 _1_ AGT --- 13 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [11] 0 1
SFs 5.534904583 _1_ 13 [1 -> 2] 1(0) to 2
r 5.540513471 _2_ AGT --- 13 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [11] 1 1

```
s 5.901087975 _1_ AGT  --- 14 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [12] 0 1
SFs 5.901087975 _1_ 14 [1 -> 2] 1(0) to 2
r 5.907076863 _2_ AGT  --- 14 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [12] 1 1
s 6.129028495 _1_ AGT  --- 15 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [13] 0 1
SFs 6.129028495 _1_ 15 [1 -> 2] 1(0) to 2
r 6.134497383 _2_ AGT  --- 15 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [13] 1 1
s 6.421928900 _1_ AGT  --- 16 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [14] 0 1
SFs 6.421928900 _1_ 16 [1 -> 2] 1(0) to 2
r 6.427937788 _2_ AGT  --- 16 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [14] 1 1
s 6.710015170 _1_ AGT  --- 17 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [15] 0 1
SFs 6.710015170 _1_ 17 [1 -> 2] 1(0) to 2
r 6.715864058 _2_ AGT  --- 17 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [15] 1 1
s 7.039695063 _1_ AGT  --- 18 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [16] 0 1
SFs 7.039695063 _1_ 18 [1 -> 2] 1(0) to 2
r 7.045183951 _2_ AGT  --- 18 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [16] 1 1
s 7.167374931 _1_ AGT  --- 19 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [17] 0 1
SFs 7.167374931 _1_ 19 [1 -> 2] 1(0) to 2
r 7.173183819 _2_ AGT  --- 19 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [17] 1 1
s 7.310241182 _1_ AGT  --- 20 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [18] 0 1
SFs 7.310241182 _1_ 20 [1 -> 2] 1(0) to 2
r 7.316050070 _2_ AGT  --- 20 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [18] 1 1
s 7.610694711 _1_ AGT  --- 21 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [19] 0 1
SFs 7.610694711 _1_ 21 [1 -> 2] 1(0) to 2
r 7.616303599 _2_ AGT  --- 21 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [19] 1 1
s 7.703020315 _7_ AGT  --- 22 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [0] 0 2
SFESTs 7.778355938 _7_ 22 [7 -> 9] 1(1) to 4 [7 |4 9 ]
r 7.790344314 _9_ AGT  --- 22 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [0] 2 2
s 7.812679631 _1_ AGT  --- 27 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [20] 0 1
SFs 7.812679631 _1_ 27 [1 -> 2] 1(0) to 2
r 7.818328519 _2_ AGT  --- 27 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [20] 1 1
s 7.895942540 _7_ AGT  --- 28 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [1] 0 2
SFESTs 7.895942540 _7_ 28 [7 -> 9] 1(1) to 4 [7 |4 9 ]
r 7.907966915 _9_ AGT  --- 28 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [1] 2 2
s 8.066985329 _1_ AGT  --- 29 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [21] 0 1
SFESTs 8.066985329 _1_ 29 [1 -> 2] 1(1) to 2 [1 |2 ]
r 8.072942217 _2_ AGT  --- 29 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [21] 1 1
s 8.188474616 _7_ AGT  --- 30 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [2] 0 2
SFESTs 8.188474616 _7_ 30 [7 -> 9] 1(1) to 4 [7 |4 9 ]
r 8.200358989 _9_ AGT  --- 30 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [2] 2 2
s 8.214207399 _1_ AGT  --- 31 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [22] 0 1
SFs 8.214207399 _1_ 31 [1 -> 2] 1(0) to 2
r 8.219716287 _2_ AGT  --- 31 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [22] 1 1
s 8.319263366 _7_ AGT  --- 32 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [3] 0 2
SFs 8.319263366 _7_ 32 [7 -> 9] 1(0) to 4
SFf 8.325093540 _4_ 32 [7 -> 9] 1 to 9
r 8.331227738 _9_ AGT  --- 32 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [3] 2 2
```

s 8.418917865 _1_ AGT --- 33 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [23] 0 1
SFs 8.418917865 _1_ 33 [1 -> 2] 1(0) to 2
r 8.424446753 _2_ AGT --- 33 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [23] 1 1
s 8.457815941 _7_ AGT --- 34 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [4] 0 2
SFs 8.457815941 _7_ 34 [7 -> 9] 1(0) to 4
SFf 8.463646114 _4_ 34 [7 -> 9] 1 to 9
r 8.469820312 _9_ AGT --- 34 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [4] 2 2
s 8.685945301 _7_ AGT --- 35 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [5] 0 2
SFs 8.685945301 _7_ 35 [7 -> 9] 1(0) to 4
SFf 8.691875473 _4_ 35 [7 -> 9] 1 to 9
r 8.697929670 _9_ AGT --- 35 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [5] 2 2
s 8.732314987 _1_ AGT --- 36 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [24] 0 1
SFs 8.732314987 _1_ 36 [1 -> 2] 1(0) to 2
r 8.738243875 _2_ AGT --- 36 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [24] 1 1
s 9.021993570 _7_ AGT --- 37 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [6] 0 2
SFs 9.021993570 _7_ 37 [7 -> 9] 1(0) to 4
SFf 9.028023742 _4_ 37 [7 -> 9] 1 to 9
r 9.034257937 _9_ AGT --- 37 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [6] 2 2
s 9.095078328 _1_ AGT --- 38 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [25] 0 1
SFs 9.095078328 _1_ 38 [1 -> 2] 1(0) to 2
r 9.101107216 _2_ AGT --- 38 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [25] 1 1
s 9.165406433 _7_ AGT --- 39 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [7] 0 2
SFs 9.165406433 _7_ 39 [7 -> 9] 1(0) to 4
SFf 9.171236605 _4_ 39 [7 -> 9] 1 to 9
r 9.177430800 _9_ AGT --- 39 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [7] 2 2
s 9.244827872 _1_ AGT --- 40 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [26] 0 1
SFs 9.244827872 _1_ 40 [1 -> 2] 1(0) to 2
r 9.250576760 _2_ AGT --- 40 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [26] 1 1
s 9.502148551 _7_ AGT --- 41 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [8] 0 2
SFs 9.502148551 _7_ 41 [7 -> 9] 1(0) to 4
SFf 9.507758721 _4_ 41 [7 -> 9] 1 to 9
r 9.513972915 _9_ AGT --- 41 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [8] 2 2
s 9.554483044 _1_ AGT --- 42 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [27] 0 1
SFs 9.554483044 _1_ 42 [1 -> 2] 1(0) to 2
r 9.560111932 _2_ AGT --- 42 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [27] 1 1
s 9.647638638 _7_ AGT --- 43 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [9] 0 2
SFs 9.647638638 _7_ 43 [7 -> 9] 1(0) to 4
SFf 9.653368807 _4_ 43 [7 -> 9] 1 to 9
r 9.659623000 _9_ AGT --- 43 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [9] 2 2
s 9.737078157 _1_ AGT --- 44 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [28] 0 1
SFs 9.737078157 _1_ 44 [1 -> 2] 1(0) to 2
r 9.742647045 _2_ AGT --- 44 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [28] 1 1
s 9.797617638 _7_ AGT --- 45 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [10] 0 2
SFs 9.797617638 _7_ 45 [7 -> 9] 1(0) to 4
SFf 9.803227807 _4_ 45 [7 -> 9] 1 to 9
r 9.809421999 _9_ AGT --- 45 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [10] 2 2

s 9.906409876 _1_ AGT --- 46 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [29] 0 1
SFs 9.906409876 _1_ 46 [1 -> 2] 1(0) to 2
r 9.912358764 _2_ AGT --- 46 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [29] 1 1
s 10.043340533 _7_ AGT --- 47 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [11] 0 2
SFs 10.043340533 _7_ 47 [7 -> 9] 1(0) to 4
SFf 10.049330702 _4_ 47 [7 -> 9] 1 to 9
r 10.055144893 _9_ AGT --- 47 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [11] 2 2
s 10.095347953 _1_ AGT --- 48 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [30] 0 1
SFs 10.095347953 _1_ 48 [1 -> 2] 1(0) to 2
r 10.101276841 _2_ AGT --- 48 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [30] 1 1
s 10.340283066 _1_ AGT --- 49 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [31] 0 1
SFs 10.340283066 _1_ 49 [1 -> 2] 1(0) to 2
s 10.342592296 _7_ AGT --- 50 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [12] 0 2
SFs 10.342592296 _7_ 50 [7 -> 9] 1(0) to 4
r 10.345991954 _2_ AGT --- 49 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [31] 1 1
SFf 10.352391199 _4_ 50 [7 -> 9] 1 to 9
r 10.358385389 _9_ AGT --- 50 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [12] 2 2
s 10.489047854 _7_ AGT --- 51 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [13] 0 2
SFs 10.489047854 _7_ 51 [7 -> 9] 1(0) to 4
SFf 10.494558021 _4_ 51 [7 -> 9] 1 to 9
r 10.500872211 _9_ AGT --- 51 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [13] 2 2
s 10.621954157 _7_ AGT --- 52 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [14] 0 2
SFs 10.621954157 _7_ 52 [7 -> 9] 1(0) to 4
SFf 10.627584323 _4_ 52 [7 -> 9] 1 to 9
r 10.633698513 _9_ AGT --- 52 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [14] 2 2
s 10.708187178 _1_ AGT --- 53 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [32] 0 1
SFs 10.708187178 _1_ 53 [1 -> 2] 1(0) to 2
r 10.713776066 _2_ AGT --- 53 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [32] 1 1
s 10.943061561 _7_ AGT --- 54 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [15] 0 2
SFs 10.943061561 _7_ 54 [7 -> 9] 1(0) to 4
SFf 10.948551726 _4_ 54 [7 -> 9] 1 to 9
r 10.954825915 _9_ AGT --- 54 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [15] 2 2
s 10.973041888 _1_ AGT --- 55 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [33] 0 1
SFs 10.973041888 _1_ 55 [1 -> 2] 1(0) to 2
r 10.978790776 _2_ AGT --- 55 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [33] 1 1
s 11.075889833 _7_ AGT --- 56 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [16] 0 2
SFs 11.075889833 _7_ 56 [7 -> 9] 1(0) to 4
SFf 11.081699998 _4_ 56 [7 -> 9] 1 to 9
r 11.087854186 _9_ AGT --- 56 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [16] 2 2
s 11.119877551 _1_ AGT --- 57 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [34] 0 1
SFs 11.119877551 _1_ 57 [1 -> 2] 1(0) to 2
r 11.125826439 _2_ AGT --- 57 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [34] 1 1
s 11.418527148 _1_ AGT --- 58 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [35] 0 1
SFs 11.418527148 _1_ 58 [1 -> 2] 1(0) to 2
r 11.424536036 _2_ AGT --- 58 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [35] 1 1
s 11.436723716 _7_ AGT --- 59 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [17] 0 2

```
SFs 11.436723716 _7_ 59 [7 -> 9] 1(0) to 4
SFf 11.442533880 _4_ 59 [7 -> 9] 1 to 9
r 11.448708067 _9_ AGT  --- 59 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [17] 2 2
s 11.596267592 _1_ AGT  --- 60 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [36] 0 1
SFs 11.596267592 _1_ 60 [1 -> 2] 1(0) to 2
r 11.602216480 _2_ AGT  --- 60 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [36] 1 1
s 11.754252739 _7_ AGT  --- 61 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [18] 0 2
SFs 11.754252739 _7_ 61 [7 -> 9] 1(0) to 4
SFf 11.760242902 _4_ 61 [7 -> 9] 1 to 9
r 11.766357087 _9_ AGT  --- 61 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [18] 2 2
s 11.899763054 _1_ AGT  --- 62 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [37] 0 1
SFs 11.899763054 _1_ 62 [1 -> 2] 1(0) to 2
r 11.905531942 _2_ AGT  --- 62 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [37] 1 1
s 12.037164376 _7_ AGT  --- 63 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [19] 0 2
SFs 12.037164376 _7_ 63 [7 -> 9] 1(0) to 4
SFf 12.042754538 _4_ 63 [7 -> 9] 1 to 9
r 12.048668723 _9_ AGT  --- 63 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [19] 2 2
s 12.134475313 _1_ AGT  --- 64 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [38] 0 1
SFs 12.134475313 _1_ 64 [1 -> 2] 1(0) to 2
r 12.140224201 _2_ AGT  --- 64 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [38] 1 1
s 12.407857671 _7_ AGT  --- 65 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [20] 0 2
SFs 12.407857671 _7_ 65 [7 -> 9] 1(0) to 4
SFf 12.413487831 _4_ 65 [7 -> 9] 1 to 9
r 12.419722015 _9_ AGT  --- 65 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [20] 2 2
s 12.436067632 _1_ AGT  --- 66 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [39] 0 1
SFs 12.436067632 _1_ 66 [1 -> 2] 1(0) to 2
r 12.441836520 _2_ AGT  --- 66 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [39] 1 1
s 12.633043785 _7_ AGT  --- 67 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [21] 0 2
SFs 12.633043785 _7_ 67 [7 -> 9] 1(0) to 4
SFf 12.639073945 _4_ 67 [7 -> 9] 1 to 9
r 12.645328128 _9_ AGT  --- 67 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [21] 2 2
s 12.722613707 _1_ AGT  --- 68 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [40] 0 1
SFs 12.722613707 _1_ 68 [1 -> 2] 1(0) to 2
r 12.728642595 _2_ AGT  --- 68 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [40] 1 1
s 12.981666940 _7_ AGT  --- 69 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [22] 0 2
SFs 12.981666940 _7_ 69 [7 -> 9] 1(0) to 4
SFf 12.987417099 _4_ 69 [7 -> 9] 1 to 9
r 12.993671280 _9_ AGT  --- 69 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [22] 2 2
s 13.049803144 _1_ AGT  --- 70 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [41] 0 1
SFs 13.049803144 _1_ 70 [1 -> 2] 1(0) to 2
r 13.055812032 _2_ AGT  --- 70 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [41] 1 1
s 13.226313297 _7_ AGT  --- 71 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [23] 0 2
SFESTs 13.226313297 _7_ 71 [7 -> 9] 1(1) to 4 [7 |4 9 ]
r 13.238157636 _9_ AGT  --- 71 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [23] 2 2
s 13.357956284 _1_ AGT  --- 72 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [42] 0 1
SFESTs 13.357956284 _1_ 72 [1 -> 2] 1(1) to 2 [1 |2 ]
```

```
r 13.364013172 _2_ AGT  --- 72 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [42] 1 1
s 13.387054785 _7_ AGT  --- 73 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [24] 0 2
SFs 13.387054785 _7_ 73 [7 -> 9] 1(0) to 4
SFf 13.392704942 _4_ 73 [7 -> 9] 1 to 9
r 13.398699122 _9_ AGT  --- 73 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [24] 2 2
s 13.625727557 _1_ AGT  --- 74 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [43] 0 1
SFs 13.625727557 _1_ 74 [1 -> 2] 1(0) to 2
r 13.631416446 _2_ AGT  --- 74 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [43] 1 1
s 13.636291369 _7_ AGT  --- 75 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [25] 0 2
SFs 13.636291369 _7_ 75 [7 -> 9] 1(0) to 4
SFf 13.642161526 _4_ 75 [7 -> 9] 1 to 9
r 13.648115705 _9_ AGT  --- 75 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [25] 2 2
s 13.868560499 _7_ AGT  --- 76 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [26] 0 2
SFs 13.868560499 _7_ 76 [7 -> 9] 1(0) to 4
SFf 13.874230655 _4_ 76 [7 -> 9] 1 to 9
r 13.880144834 _9_ AGT  --- 76 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [26] 2 2
s 13.883775640 _1_ AGT  --- 77 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [44] 0 1
SFs 13.883775640 _1_ 77 [1 -> 2] 1(0) to 2
r 13.889284528 _2_ AGT  --- 77 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [44] 1 1
s 14.034843877 _7_ AGT  --- 78 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [27] 0 2
SFs 14.034843877 _7_ 78 [7 -> 9] 1(0) to 4
SFf 14.040734032 _4_ 78 [7 -> 9] 1 to 9
r 14.046628210 _9_ AGT  --- 78 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [27] 2 2
s 14.062668235 _1_ AGT  --- 79 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [45] 0 1
SFs 14.062668235 _1_ 79 [1 -> 2] 1(0) to 2
r 14.068117123 _2_ AGT  --- 79 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [45] 1 1
s 14.217900663 _7_ AGT  --- 80 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [28] 0 2
SFs 14.217900663 _7_ 80 [7 -> 9] 1(0) to 4
SFf 14.223910817 _4_ 80 [7 -> 9] 1 to 9
r 14.229664994 _9_ AGT  --- 80 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [28] 2 2
s 14.334261133 _1_ AGT  --- 81 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [46] 0 1
SFs 14.334261133 _1_ 81 [1 -> 2] 1(0) to 2
r 14.339830022 _2_ AGT  --- 81 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [46] 1 1
s 14.376353704 _7_ AGT  --- 82 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [29] 0 2
SFs 14.376353704 _7_ 82 [7 -> 9] 1(0) to 4
SFf 14.381963858 _4_ 82 [7 -> 9] 1 to 9
r 14.387838035 _9_ AGT  --- 82 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [29] 2 2
s 14.602537452 _1_ AGT  --- 83 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [47] 0 1
SFs 14.602537452 _1_ 83 [1 -> 2] 1(0) to 2
r 14.608426341 _2_ AGT  --- 83 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [47] 1 1
s 14.662238752 _7_ AGT  --- 84 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [30] 0 2
SFs 14.662238752 _7_ 84 [7 -> 9] 1(0) to 4
SFf 14.668128905 _4_ 84 [7 -> 9] 1 to 9
r 14.674023081 _9_ AGT  --- 84 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [30] 2 2
s 14.872370010 _7_ AGT  --- 85 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [31] 0 2
SFs 14.872370010 _7_ 85 [7 -> 9] 1(0) to 4
```

SFf 14.878280163 _4_ 85 [7 -> 9] 1 to 9
r 14.884494337 _9_ AGT --- 85 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [31] 2 2
s 14.923561325 _1_ AGT --- 86 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [48] 0 1
SFs 14.923561325 _1_ 86 [1 -> 2] 1(0) to 2
r 14.929210214 _2_ AGT --- 86 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [48] 1 1
s 15.230720276 _7_ AGT --- 87 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [32] 0 2
SFs 15.230720276 _7_ 87 [7 -> 9] 1(0) to 4
SFf 15.236290427 _4_ 87 [7 -> 9] 1 to 9
r 15.242144601 _9_ AGT --- 87 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [32] 2 2
s 15.289016845 _1_ AGT --- 88 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [49] 0 1
SFs 15.289016845 _1_ 88 [1 -> 2] 1(0) to 2
r 15.294765733 _2_ AGT --- 88 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [49] 1 1
s 15.413768229 _7_ AGT --- 89 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [33] 0 2
SFs 15.413768229 _7_ 89 [7 -> 9] 1(0) to 4
SFf 15.419778380 _4_ 89 [7 -> 9] 1 to 9
r 15.425912553 _9_ AGT --- 89 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [33] 2 2
s 15.573425374 _7_ AGT --- 90 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [34] 0 2
SFs 15.573425374 _7_ 90 [7 -> 9] 1(0) to 4
SFf 15.579275525 _4_ 90 [7 -> 9] 1 to 9
r 15.585349697 _9_ AGT --- 90 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [34] 2 2
s 15.596107543 _1_ AGT --- 91 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [50] 0 1
SFs 15.596107543 _1_ 91 [1 -> 2] 1(0) to 2
r 15.602136431 _2_ AGT --- 91 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [50] 1 1
s 15.747524296 _1_ AGT --- 92 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [51] 0 1
SFs 15.747524296 _1_ 92 [1 -> 2] 1(0) to 2
r 15.753513184 _2_ AGT --- 92 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [51] 1 1
s 15.925629444 _7_ AGT --- 93 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [35] 0 2
SFs 15.925629444 _7_ 93 [7 -> 9] 1(0) to 4
SFf 15.931579594 _4_ 93 [7 -> 9] 1 to 9
r 15.937873765 _9_ AGT --- 93 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [35] 2 2
s 16.018382992 _1_ AGT --- 94 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [52] 0 1
SFs 16.018382992 _1_ 94 [1 -> 2] 1(0) to 2
r 16.023911881 _2_ AGT --- 94 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [52] 1 1
s 16.189230122 _7_ AGT --- 95 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [36] 0 2
SFs 16.189230122 _7_ 95 [7 -> 9] 1(0) to 4
SFf 16.194800271 _4_ 95 [7 -> 9] 1 to 9
r 16.200654441 _9_ AGT --- 95 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [36] 2 2
s 16.267052214 _1_ AGT --- 96 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [53] 0 1
SFs 16.267052214 _1_ 96 [1 -> 2] 1(0) to 2
r 16.272841103 _2_ AGT --- 96 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [53] 1 1
s 16.452349939 _1_ AGT --- 97 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [54] 0 1
SFs 16.452349939 _1_ 97 [1 -> 2] 1(0) to 2
r 16.458198828 _2_ AGT --- 97 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [54] 1 1
s 16.460826960 _7_ AGT --- 98 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [37] 0 2
SFs 16.460826960 _7_ 98 [7 -> 9] 1(0) to 4
SFf 16.466377108 _4_ 98 [7 -> 9] 1 to 9

r 16.472631277 _9_ AGT --- 98 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [37] 2 2
s 16.757562271 _7_ AGT --- 99 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [38] 0 2
SFs 16.757562271 _7_ 99 [7 -> 9] 1(0) to 4
s 16.761203329 _1_ AGT --- 100 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [55] 0 1
SFs 16.761203329 _1_ 100 [1 -> 2] 1(0) to 2
SFf 16.763192418 _4_ 99 [7 -> 9] 1 to 9
r 16.769126050 _2_ AGT --- 100 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [55] 1 1
r 16.775221019 _9_ AGT --- 99 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [38] 2 2
s 16.905266913 _7_ AGT --- 101 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [39] 0 2
SFs 16.905266913 _7_ 101 [7 -> 9] 1(0) to 4
SFf 16.910757059 _4_ 101 [7 -> 9] 1 to 9
r 16.916811227 _9_ AGT --- 101 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [39] 2 2
s 17.094696251 _1_ AGT --- 102 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [56] 0 1
SFs 17.094696251 _1_ 102 [1 -> 2] 1(0) to 2
r 17.100625140 _2_ AGT --- 102 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [56] 1 1
s 17.188769677 _7_ AGT --- 103 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [40] 0 2
SFs 17.188769677 _7_ 103 [7 -> 9] 1(0) to 4
SFf 17.194519823 _4_ 103 [7 -> 9] 1 to 9
r 17.200773989 _9_ AGT --- 103 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [40] 2 2
s 17.414126652 _1_ AGT --- 104 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [57] 0 1
SFs 17.414126652 _1_ 104 [1 -> 2] 1(0) to 2
r 17.419935541 _2_ AGT --- 104 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [57] 1 1
s 17.553383911 _7_ AGT --- 105 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [41] 0 2
SFs 17.553383911 _7_ 105 [7 -> 9] 1(0) to 4
SFf 17.559374055 _4_ 105 [7 -> 9] 1 to 9
r 17.565688221 _9_ AGT --- 105 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [41] 2 2
s 17.581277550 _1_ AGT --- 106 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [58] 0 1
SFs 17.581277550 _1_ 106 [1 -> 2] 1(0) to 2
r 17.587086439 _2_ AGT --- 106 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [58] 1 1
s 17.877877666 _1_ AGT --- 107 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [59] 0 1
SFs 17.877877666 _1_ 107 [1 -> 2] 1(0) to 2
r 17.883866555 _2_ AGT --- 107 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [59] 1 1
s 17.924410390 _7_ AGT --- 108 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [42] 0 2
SFs 17.924410390 _7_ 108 [7 -> 9] 1(0) to 4
SFf 17.930060533 _4_ 108 [7 -> 9] 1 to 9
r 17.936014697 _9_ AGT --- 108 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [42] 2 2
s 18.156543324 _1_ AGT --- 109 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [60] 0 1
SFs 18.156543324 _1_ 109 [1 -> 2] 1(0) to 2
r 18.162032213 _2_ AGT --- 109 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [60] 1 1
s 18.170550009 _7_ AGT --- 110 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [43] 0 2
SFs 18.170550009 _7_ 110 [7 -> 9] 1(0) to 4
SFf 18.176020152 _4_ 110 [7 -> 9] 1 to 9
r 18.182114315 _9_ AGT --- 110 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [43] 2 2
s 18.309756793 _1_ AGT --- 111 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [61] 0 1
SFs 18.309756793 _1_ 111 [1 -> 2] 1(0) to 2
r 18.315625682 _2_ AGT --- 111 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [61] 1 1

s 18.341510395 _7_ AGT --- 112 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [44] 0 2
SFESTs 18.341510395 _7_ 112 [7 -> 9] 1(1) to 4 [7 |4 9]
r 18.353914699 _9_ AGT --- 112 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [44] 2 2
s 18.638118586 _7_ AGT --- 113 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [45] 0 2
SFs 18.638118586 _7_ 113 [7 -> 9] 1(0) to 4
SFf 18.643648727 _4_ 113 [7 -> 9] 1 to 9
r 18.649802889 _9_ AGT --- 113 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [45] 2 2
s 18.675455455 _1_ AGT --- 114 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [62] 0 1
SFESTs 18.675455455 _1_ 114 [1 -> 2] 1(1) to 2 [1 |2]
r 18.681172344 _2_ AGT --- 114 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [62] 1 1
s 18.893493800 _1_ AGT --- 115 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [63] 0 1
SFs 18.893493800 _1_ 115 [1 -> 2] 1(0) to 2
r 18.899402690 _2_ AGT --- 115 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [63] 1 1
s 18.926066265 _7_ AGT --- 116 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [46] 0 2
SFs 18.926066265 _7_ 116 [7 -> 9] 1(0) to 4
SFf 18.931836405 _4_ 116 [7 -> 9] 1 to 9
r 18.937870565 _9_ AGT --- 116 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [46] 2 2
s 19.069061100 _7_ AGT --- 117 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [47] 0 2
SFs 19.069061100 _7_ 117 [7 -> 9] 1(0) to 4
SFf 19.074871240 _4_ 117 [7 -> 9] 1 to 9
r 19.081105400 _9_ AGT --- 117 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [47] 2 2
s 19.227528072 _1_ AGT --- 118 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [64] 0 1
SFs 19.227528072 _1_ 118 [1 -> 2] 1(0) to 2
r 19.232996962 _2_ AGT --- 118 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [64] 1 1
s 19.432217121 _7_ AGT --- 119 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [48] 0 2
SFs 19.432217121 _7_ 119 [7 -> 9] 1(0) to 4
SFf 19.438027259 _4_ 119 [7 -> 9] 1 to 9
r 19.444061418 _9_ AGT --- 119 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [48] 2 2
s 19.562840875 _1_ AGT --- 120 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [65] 0 1
SFs 19.562840875 _1_ 120 [1 -> 2] 1(0) to 2
r 19.568489765 _2_ AGT --- 120 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [65] 1 1
s 19.608498987 _7_ AGT --- 121 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [49] 0 2
SFs 19.608498987 _7_ 121 [7 -> 9] 1(0) to 4
SFf 19.614129125 _4_ 121 [7 -> 9] 1 to 9
r 19.620383283 _9_ AGT --- 121 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [49] 2 2
s 19.733019939 _1_ AGT --- 122 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [66] 0 1
SFs 19.733019939 _1_ 122 [1 -> 2] 1(0) to 2
r 19.738488829 _2_ AGT --- 122 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [66] 1 1
s 19.921181539 _1_ AGT --- 123 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [67] 0 1
SFs 19.921181539 _1_ 123 [1 -> 2] 1(0) to 2
r 19.926790429 _2_ AGT --- 123 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [67] 1 1
s 19.949305024 _7_ AGT --- 124 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [50] 0 2
SFs 19.949305024 _7_ 124 [7 -> 9] 1(0) to 4
SFf 19.954915161 _4_ 124 [7 -> 9] 1 to 9
r 19.960649318 _9_ AGT --- 124 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [50] 2 2
s 20.225351374 _7_ AGT --- 125 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [51] 0 2

SFs 20.225351374 _7_ 125 [7 -> 9] 1(0) to 4
SFf 20.230801510 _4_ 125 [7 -> 9] 1 to 9
r 20.236755666 _9_ AGT --- 125 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [51] 2 2
s 20.277464527 _1_ AGT --- 126 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [68] 0 1
SFs 20.277464527 _1_ 126 [1 -> 2] 1(0) to 2
r 20.283033417 _2_ AGT --- 126 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [68] 1 1
s 20.485484684 _8_ AGT --- 127 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [0] 0 2
SFESTs 20.485484684 _8_ 127 [8 -> 9] 1(1) to 7 [8 |7 4 9]
r 20.508635931 _9_ AGT --- 127 cbr 512 [13a 9 4 800] ------- [8:1 9:1 30 9] [0] 3 2
s 20.565633698 _1_ AGT --- 129 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [69] 0 1
SFs 20.565633698 _1_ 129 [1 -> 2] 1(0) to 2
r 20.571402588 _2_ AGT --- 129 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [69] 1 1
s 20.578059987 _7_ AGT --- 130 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [52] 0 2
SFs 20.578059987 _7_ 130 [7 -> 9] 1(0) to 4
SFf 20.583770123 _4_ 130 [7 -> 9] 1 to 9
r 20.589624277 _9_ AGT --- 130 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [52] 2 2
s 20.779794438 _8_ AGT --- 131 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [1] 0 2
SFESTs 20.779794438 _8_ 131 [8 -> 9] 3(1) to 4 [8 |4 9]
r 20.791738533 _9_ AGT --- 131 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [1] 2 2
s 20.818498010 _7_ AGT --- 132 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [53] 0 2
SFs 20.818498010 _7_ 132 [7 -> 9] 1(0) to 4
SFf 20.823948145 _4_ 132 [7 -> 9] 1 to 9
r 20.830002298 _9_ AGT --- 132 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [53] 2 2
s 20.902636673 _1_ AGT --- 133 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [70] 0 1
SFs 20.902636673 _1_ 133 [1 -> 2] 1(0) to 2
r 20.908385564 _2_ AGT --- 133 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [70] 1 1
s 20.991303929 _7_ AGT --- 134 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [54] 0 2
SFs 20.991303929 _7_ 134 [7 -> 9] 1(0) to 4
SFf 20.997294063 _4_ 134 [7 -> 9] 1 to 9
r 21.003368216 _9_ AGT --- 134 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [54] 2 2
s 21.086546064 _8_ AGT --- 135 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [2] 0 2
SFESTs 21.086546064 _8_ 135 [8 -> 9] 3(1) to 4 [8 |4 9]
r 21.098390158 _9_ AGT --- 135 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [2] 2 2
s 21.249910931 _1_ AGT --- 136 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [71] 0 1
SFs 21.249910931 _1_ 136 [1 -> 2] 1(0) to 2
r 21.255659821 _2_ AGT --- 136 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [71] 1 1
s 21.266914768 _8_ AGT --- 137 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [3] 0 2
SFESTs 21.266914768 _8_ 137 [8 -> 9] 3(1) to 4 [8 |4 9]
r 21.279058861 _9_ AGT --- 137 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [3] 2 2
s 21.283169746 _7_ AGT --- 138 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [55] 0 2
SFs 21.283169746 _7_ 138 [7 -> 9] 1(0) to 4
SFf 21.289039879 _4_ 138 [7 -> 9] 1 to 9
r 21.294814032 _9_ AGT --- 138 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [55] 2 2
s 21.429973804 _8_ AGT --- 139 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [4] 0 2
SFs 21.429973804 _8_ 139 [8 -> 9] 3(0) to 4
SFf 21.435803743 _4_ 139 [8 -> 9] 3 to 9

r 21.441657895 _9_ AGT --- 139 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [4] 2 2
s 21.583234841 _8_ AGT --- 140 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [5] 0 2
SFs 21.583234841 _8_ 140 [8 -> 9] 3(0) to 4
s 21.583247102 _7_ AGT --- 141 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [56] 0 2
SFs 21.583247102 _7_ 141 [7 -> 9] 1(0) to 4
SFf 21.588917234 _4_ 141 [7 -> 9] 1 to 9
r 21.594671385 _9_ AGT --- 141 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [56] 2 2
SFf 21.601590639 _4_ 140 [8 -> 9] 3 to 9
r 21.607344790 _9_ AGT --- 140 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [5] 2 2
s 21.609791004 _1_ AGT --- 142 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [72] 0 1
SFs 21.609791004 _1_ 142 [1 -> 2] 1(0) to 2
r 21.615279895 _2_ AGT --- 142 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [72] 1 1
s 21.736791129 _1_ AGT --- 143 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [73] 0 1
SFs 21.736791129 _1_ 143 [1 -> 2] 1(0) to 2
r 21.742660019 _2_ AGT --- 143 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [73] 1 1
s 21.818793248 _8_ AGT --- 144 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [6] 0 2
SFs 21.818793248 _8_ 144 [8 -> 9] 3(0) to 4
SFf 21.824343185 _4_ 144 [8 -> 9] 3 to 9
r 21.830157336 _9_ AGT --- 144 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [6] 2 2
s 21.870357512 _7_ AGT --- 145 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [57] 0 2
SFs 21.870357512 _7_ 145 [7 -> 9] 1(0) to 4
SFf 21.876007643 _4_ 145 [7 -> 9] 1 to 9
r 21.881961793 _9_ AGT --- 145 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [57] 2 2
s 21.995439626 _1_ AGT --- 146 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [74] 0 1
SFs 21.995439626 _1_ 146 [1 -> 2] 1(0) to 2
r 22.001328517 _2_ AGT --- 146 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [74] 1 1
s 22.036683613 _8_ AGT --- 147 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [7] 0 2
SFs 22.036683613 _8_ 147 [8 -> 9] 3(0) to 4
SFf 22.042273550 _4_ 147 [8 -> 9] 3 to 9
r 22.048147699 _9_ AGT --- 147 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [7] 2 2
s 22.173308439 _7_ AGT --- 148 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [58] 0 2
SFs 22.173308439 _7_ 148 [7 -> 9] 1(0) to 4
SFf 22.178878570 _4_ 148 [7 -> 9] 1 to 9
r 22.185152719 _9_ AGT --- 148 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [58] 2 2
s 22.288119304 _1_ AGT --- 149 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [75] 0 1
SFs 22.288119304 _1_ 149 [1 -> 2] 1(0) to 2
r 22.293648195 _2_ AGT --- 149 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [75] 1 1
s 22.298349195 _7_ AGT --- 150 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [59] 0 2
SFs 22.298349195 _7_ 150 [7 -> 9] 1(0) to 4
SFf 22.304199325 _4_ 150 [7 -> 9] 1 to 9
r 22.310173474 _9_ AGT --- 150 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [59] 2 2
s 22.355023199 _8_ AGT --- 151 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [8] 0 2
r 22.367247282 _9_ AGT --- 151 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [8] 2 2
s 22.525674697 _8_ AGT --- 152 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [9] 0 2
SFs 22.525674697 _8_ 152 [8 -> 9] 3(0) to 4
SFf 22.531484632 _4_ 152 [8 -> 9] 3 to 9

```
r 22.537518779 _9_ AGT  --- 152 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [9] 2 2
s 22.548804375 _1_ AGT  --- 153 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [76] 0 1
SFs 22.548804375 _1_ 153 [1 -> 2] 1(0) to 2
r 22.554433266 _2_ AGT  --- 153 cbr 512 [13a 2 1 800] ------- [1:0 2:0 32 2] [76] 1 1
s 22.605255774 _7_ AGT  --- 154 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [60] 0 2
SFs 22.605255774 _7_ 154 [7 -> 9] 1(0) to 4
SFf 22.611145903 _4_ 154 [7 -> 9] 1 to 9
r 22.617340050 _9_ AGT  --- 154 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [60] 2 2
s 22.779821775 _8_ AGT  --- 155 cbr 512 [0 0 0 0] ------- [8:1 9:1 32 0] [10] 0 2
SFs 22.779821775 _8_ 155 [8 -> 9] 3(0) to 4
SFf 22.785571708 _4_ 155 [8 -> 9] 3 to 9
r 22.791525855 _9_ AGT  --- 155 cbr 512 [13a 9 4 800] ------- [8:1 9:1 31 9] [10] 2 2
s 22.800257224 _7_ AGT  --- 156 cbr 512 [0 0 0 0] ------- [7:2 9:0 32 0] [61] 0 2
SFs 22.800257224 _7_ 156 [7 -> 9] 1(0) to 4
SFf 22.805887353 _4_ 156 [7 -> 9] 1 to 9
r 22.812061500 _9_ AGT  --- 156 cbr 512 [13a 9 4 800] ------- [7:2 9:0 31 9] [61] 2 2
s 22.869852140 _1_ AGT  --- 157 cbr 512 [0 0 0 0] ------- [1:0 2:0 32 0] [77] 0 1
SFs 22.869852140 _1_ 157 [1 -> 2] 1(0) to 2
.....
D 91.717812082 _9_ RTR TOUT 1149 cbr 532 [0 0 0 0] ------- [9:2 10:0 32 0] [0] 0 0
```

We can see that the response time under SDDSR is faster compared to that of DSR. The difference is made during the route setup process, because it takes DSR more time to setup the routes since it uses source routing during the route setup. SDDSR also showed better performance concerning packet delivery. We can see that the only packets dropped by using SDDSR are due to TTL settings. In other words we can avoid those dropped packets by changing the TTL to cope with network.

References:

[1] Elizabeth M. Royer and Chai-Keong Toh,"A Review of Current Routing Protocols for Ad hoc Mobile Wireless Networks," IEEE Personal Communications, Vol. 6, No. 2. (1999), pp. 46-55.

[2] Christian Bettstetter, "Topology Properties of Ad Hoc Networks with Random Waypoint Mobility," ACM SIGMOBILE Mobile Computing and Communications Review, Vol. 7, Issue 3, pages 50-52, July 2003.

[3] C. E. Perkins and E. M. Royer, "Ad-hoc On-Demand Distance Vector Routing," Mobile Comp. Sys. and Apps., Feb. 1999, pp. 90-100.

[4] David B. Johnson, David A. Maltz, and Josh Broch, "DSR: The Dynamic Source Routing Protocol for Multihop Wireless Ad Hoc Networks," Ad Hoc Networking, edited by Charles E. Perkins, Chapter 5, pages 139- 172. Addison-Wesley, 2001.

[5] Broch, D. B. Johnson, and D. A. Maltz, "The Dynamic Source Routing protocol for Mobile Ad Hoc Networks," IETF Internet draft, draft-ietf " manet-dsr-01.txt, Dec. 1998.

[6] M. S. Corson and A. Ephremides, "A Distributed Routing Algorithm for Mobile Wireless Networks," ACM Baltzer Wireless Networks J., vol. 1, no. 1, Feb. 1995, pp. 61-81.

[7] C-K. Toh, "A Novel Distributed Routing Protocol To Support Ad-Hoc Mobile Computing," Proc. 1996 IEEE 15th Annual Int'l. . Phoenix Conf. Comp, and Commun., Mar. 1996, pp. 480-86.

[8] C. E. Perkins and E. M. Royer, "Ad Hoc On Demand Distance Vector (AODV) Routing," IETF Internet draft, draft-ietf-manet-aodv-02.txt, Nov. 1998.

[9] C. Perkins, E. Belding-Royer, and S. Das, "Ad hoc On-Demand Distance Vector (AODV) Routing," IETF RFC 3561, July 2003.

[10] D. Johnson and ll. Malt.z, "Dynamic Source Routing in Ad Hoc Wireless Networks," On Computer Communications Review- Proceedings of SIGCOMM '96, Aug.1996.

[11] C. E. Perkins and E. M. Royer, "Ad Hoc On Demand Distance Vector (AODV) Routing," draft-ietf-manet-aodv-02.txt, Nov. 1998.

[12] Josh Broch, David Johnson, and David Maltz, "The Dynamic Source Routing Protocol for Mobile Ad-hoc Networks," http://www.ietf.org/internet-drafts/ draft-ietf-manet-dsrOl.txt, Dec 1998. IETF Internet Draft.

[13] S. Das, C. Perkins, and E. Royer, "Performance Comparison of Two On-demand Routing Protocols for Ad hoc Networks," In Proc. of the 2000 IEEE Computer and Communications Societies Conference on Computer Communications (INFOCOM-00), pages 3-12, 2000.

[14] S.R. Das, R. Castaneda, J. Yan, and R. Sengupta, "Comparative Performance Evaluation of Routing Protocols for Mobile, Ad hoc Networks," In 7th Int. Conf. on Computer Communications and Networks (IC3N), pages 153-161, October, 1998.

[15] Kevin Fall and Kannan Varadhan . "NS Notes and Documentation," 1999. Available from http://www-mash.cs.berkeley.edu/ns/.

[16] Per Johansson, Tony Larsson, Nicklas Hedman, and Bartosz Mielczarek, "Routing protocols for mobile ad-hoc networks - a comparative performance analysis," In Proceedings of the 5th International Conference on Mobile Computing and Networking (ACMMOBICOM'99), August 1999.

[17] D. Johnson and D. Maltz, "Dynamic Source Routing in Ad hoc Wireless Networks.," In T. Imielinski and H. Korth, editors, Mobile computing, chapter 5. Kluwer Academic, 1996.

[18] David Maltz, Josh Broch, Jorjeta Jetcheva, and David Johnson, "The Effects of On-demand Behavior in Routing Protocols for Multihop Wireless Ad hoc Networks," IEEE Journal on Selected Areas in Communication, 1999.

[19] Charles Perkins and Elizabeth Royer, "Ad hoc On-demand Distance Vector Routing," In Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications, pages 90-100, Feb 1999.

[20] S. Murthy and 1. J. Garcia-Luna-Aceves, "An Efficient Routing Protocol for Wireless Networks," ACM Mobile Networks and App. J., Special Issue on Routing in Mobile Communication Networks, Oct. 1996, pp. 183-97.

[21] Pablo Brenner, "A Technical Tutorial on the IEEE 802.11 Standard" BreezeCom, 1997.

[22] Tracy Camp, Jeff Boleng, and Vanessa Davies, "A Survey of Mobility Models for Ad Hoc Network Research," Wireless Communication & Mobile Computing (WCMC): Special Issue on Mobile Ad Hoc Networking: Research, Trends and Applications, Vol.2, No.5, pp.483-502, 2002.

[23] Mesut Gunes and Otto Spaniol, "Routing Algorithms for Mobile Multi-Hop Ad Hoc Networks," International Workshop NGNT, pages 20-24, 2002.

[24] Y.C. Hu and D.B. Johnson, "Caching Strategies in On-Demand Routing Protocols for Wireless Ad Hoc Networks," ACM 6th International Conference on Mobile Computing and Networking (MobiCom 2000), pages 231 – 242, August 2000.

[25] Yingji Zhong and Dongfeng Yuan, "Dynamic Source Routing Protocol for Wireless Ad Hoc Networks in Special Scenario using Location Information," Communication Technology Proceedings, ICCT, International Conference, Vol.2, pages 1287-1290, April 2003.

[26] Justin A. Boyan and Michael L. Littman, "Packet routing in dynamically changing networks: A reinforcement learning approach," Advances in Neural Information Processing Systems, volume 6, pages 671-678. Morgan Kaufmann, San Francisco CA, 1993.

[27] Jae Chung and Mark Claypool, "NS by Example". Available from http://www.wpi.edu.

[28] J. Raju, J. J. Garcia-Luna-Acceves, "A comparison of on-demand and table driven routingfor ad hoc networks," IEEE ICC '00, June 2000.

[29] S. Corson and J. Macker. "Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations," RFC 2002, January 1999.