TASK 1:

The text below has been divided according to possible moves. Can you determine and label each move? What expressions signal each of these moves?

E. Papapetrou and F.-N. Pavlidou, IEEE COMMUNICATIONS LETTERS, VOL. 7, NO. 10, OCTOBER 2003

A Novel Approach to Source Routing for Multi-Hop Ad Hoc Networks

I. INTRODUCTION

1-3

[1]Wireless mobile networks that operate without the need of a fixed infrastructure are widely known as ad hoc networks. [2]Due to recent technology advances, their penetration to markets worldwide has significantly increased over the last years. [3]Ad hoc networks consist of mobile hosts that move randomly in and out of each others communication range.

[4]As a result, connections between nodes are prone to sudden failures and the graph of the formatted network varies stochastically. [5]It is clear that the aforementioned context encumbers routing. [6]Therefore, the choice of a suitable routing technique is deeply affected.

[7]Traditional table-driven routing protocols [1] cannot perform efficiently in such an environment [2]. [8]The reason is that they waste the limited system resources to discover routes that are not needed.

[9]On the other hand, on-demand routing protocols [1], [3], [4] have been proposed as an effective solution to the problem.

[10]Their main advantage is that a route discovery is performed only when there is a request for communication between two network nodes. [11]Thus, the bandwidth needed for the protocol operation is minimized.

[12]One of the most representative on-demand protocols is the Dynamic Source Routing (DSR) protocol [3].

[13]It is based on source routing techniques known from IEEE 802 LAN's implementation. [14]Routes are discovered only when needed using a route discovery procedure. [15]This means that either a user request or a route break down may cause a new route discovery. [16]In this way, routing packets are minimized since only the topology changes of interest are considered.

[17]Although DSR outperforms table driven protocols, for high host mobility, frequent link failures degrade the network performance and increase the routing load [2].

[18]Another well-known on-demand protocol is AODV [4]. [19]AODV discovers paths with a procedure similar to that of DSR but without using source routing.

[20]AODV maintains tables instead of caching routes. [21]To avoid loop formation AODV makes use of sequence numbers that represent the freshness of routing information. [22]In this way, AODV manages to deliver more successfully data packets and at the same time reduce significantly the routing load involved in a route discovery phase.

[23]On the other hand, AODV fails to reduce the number of route discoveries because it does not make full use of routing information. [24]As a result the overall overhead increases as demonstrated in [2] and [5].

[25]Summarizing, it is clear that the advantage of DSR, allowing it to reduce routing load, is the use of extensive routing information stored in each node cache.

[26]On the other hand, AODV manages to avoid using stale routing information by means of sequence numbers and therefore increases delivery ratio.

[27]In this paper we will propose a new routing protocol that uses cached routes combined with sequence numbers to enhance network performance in terms of both delivery ratio and routing load.

[28]The rest of the paper is structured as follows. [29]In Section II the proposed protocol is presented in detail. [30]In Section III we present the results of a simulation study in which the new protocol is compared to DSR. [31]Finally, useful conclusions are drawn in Section IV.

TASK 2:

The words and expressions signaling each move-step sequence are shown in the introduction below. Can you determine and label each move?

Huamin Chen, Prasant Mohapatra, Overload control in QoS-aware web servers, Computer Networks 42 (2003) 119 –133

INTRODUCTION

1-2

[1]As the widespread usage of web service grows, the number of accesses to many popular web sites is **ever increasing** and occasionally reaches **the limit of their capacity** and consequently causes the servers to be **overloaded**/ [2]As a result, end users either receive a **busy signal** or **nothing** at all before the browser indicates a time-out error or the user aborts (stops) the request. [3]Subsequently, the server may **get choked** or **crash** causing **denial of services**. [4]Such **abnormality** is often regarded as the servers's **poor quality** and **compromises** their long term survivability. [5]In e-commerce applications, such server behavior could translate to **sizable revenue losses**.

[6]**Research on** overload prevention and control **has been limited** compared to the other performance improvement issues such as web caching, and load balancing in web servers. [7]These performance enhancement techniques**, however,** are **inadequate** in ensuring a busy web server from being overloaded due to the fact that the web traffic is highly **unpredictable** and **bursty** [10,15]. [8]**Proper capacity planning and forecasting methods** can prevent servers from being overloaded under controlled traffic conditions.

[9]In many web sites, especially in e-commerce, online brokers, and supply chain sites, the majority of the requests in the web traffic **are** session-based. [10]A session contains temporally and logically related request sequences from the same client. [11]Sessions can be identified either by HTTP/1.1 persistent connections [12] or from the state information within the presence of cookies [14]. [12]Sessions exhibit distinguishable features from individual requests. [13]**For example,** session integrity requires that once admitted for processing, all the following requests within a session should be honored. [14]**Similarly,** session affinity would require that requests belonging to the same session are handled by the same front-end server for security and locality reasons. [15]**These features** may **complicate** or **contradict** the research conclusions of the performance studies on web servers where the number of request completions **have been considered** as the primary performance measure. [16]**For example,** admission control on a per request basis may lead to a **large number of broken or incomplete sessions** when the system is overloaded. [17]Incomplete sessions may be equivalent to a rejected session from the users viewpoint or for most e-commerce servers. [18]Thus, performance measure based on the number of request completions may **not** be a good indication of users satisfaction (the basic purpose of web service). [19]Especially during overloads, the **disparity** between the two types of performance measures (proportion of request completion and proportion of session completion) is more enhanced. [20]Capacity planning schemes based on individual requests also have the same **deficiency**.

[21]Session integrity is a **critical metric** for commercial web service. [22]For an online retailer, the more the number of sessions completed, the more the amount of revenue that is likely to be generated. [23]The same statement cannot be made about the individual request completions. [24]Sessions that are broken or delayed at some critical stages, like checkout and shipping, could mean **loss of revenue** to the web site. [25]From the end users's perspective, this means **poor service availability**. [26]Therefore, it is more useful to use session integrity to evaluate the service availability of servers, especially during high-load periods.

[27]**In this paper, we explore** the session characteristics and their potential in overload control and prevention. [28]**A workload characterization study is done** first **to gain an insight into** the load patterns in web servers. [29]The workload characterization study was based on the server log from a popular online retailer. [30]**We found that**, despite the seemingly complication of session sequences, some statistical results can used in simplifying the session-based traffic model. [31]**Based on these results**, the session logic can be utilized for capacity planning and request scheduling of QoS-aware servers, which **improves** server's productivity. [32]Server productivity quantifies the amount of useful work done by the server. [33]Based on the session-level traffic model, **we have proposed** a dynamic weighted fair scheduling (DWFS) scheme that assign service weight to different requests of a session in a dynamic manner. [34]**We have done an experimental performance analysis** by modifying the scheduling scheme of the Apache web server. [35]The proposed DWFS scheme **provides** a performance improvement of about 50% in terms of response delay and **significantly reduces** the session abortion rate for the workload and system configuration used in the experimentation.

[36]**The rest of the paper is organized in the following manner.** [37]**Section 2 characterizes** session-based HTTP requests. [38]**Section 3 provides** capacity planning tools to prevent server overload. [39]**Section 4 proposes** a request scheduling algorithm to control server overload and improve server performance **followed by** experimental results **in Section 5**, **which proves** the feasibility and **quantifies** the performance of the proposed algorithm. [40]The related works **are discussed in Section 5**, **followed by** the concluding remarks **in Section 7**.

TASK 3:

Find, underline and label the CARS-model moves in the introduction below.

A. D'Amico, U. Mengali, and M. Morelli, Channel Estimation for the Uplink of a DS-CDMA System, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, VOL. 2, NO. 2, April 2003

I. INTRODUCTION

1Direct-sequence code-division multiple-access (DSCDMA) is a popular multiplexing technique in which several users simultaneously access the same frequency band by modulating preassigned signature waveforms [1]. **2****Recently**, it **has been adopted** as a standard for third-generation (3-G) mobile cellular networks in view of its advantages over other multiplexing methods.

3In 3-G applications, multiple propagation paths **characterize** the transmission channel. **4****An important feature** of DS-CDMA systems is its capability of resolving the multipath components

1-2 and optimally combining them in a RAKE receiver or other more sophisticated detection schemes [2]. **5**Whatever the case, accurate estimates of the differential delays and complex gains of the various propagation paths are needed to achieve reliable data detection.

6Estimating the channel parameters in the uplink of a DS-CDMA system is a challenging task. **7**The main problem is that signals transmitted by different users are asynchronous and, in consequence, not truly orthogonal. **8**This generates multiaccess interference (MAI) at the base station (BS), thereby limiting the accuracy of the channel estimates.

9The need for accurate channel estimation in the presence of MAI has led to the development of joint multiuser detection and parameter estimation techniques [3], [4]. **10**Also, blind channel estimators based on subspace methods have been proposed [5]. **11**However, these methods have long acquisition times and can hardly be used with rapidly varying channels. **12**Joint maximum likelihood (ML) estimation of the channels of all the active users is addressed in [6] and [7].

13These techniques produce excellent results but are computationally intense since they involve numerical maximizations over a large number of parameters. **14**Suboptimal schemes concentrating on a single user's channel reduce the computational complexity at the expense of some performance loss. **15**This problem is investigated in [8], where the MAI is modeled as colored Gaussian noise and the expectation–maximization (EM) algorithm [9] is used to decompose a multidimensional maximization problem into a sequence of one-dimensional searches. **16**However, this scheme is not easily adapted to the universal mobile telecommunications system (UMTS) standard as it requires a training sequence consisting of the repetition of a fixed pilot symbol, which is not envisioned in the standard. **17**Also, it is tailored for rectangular pulses whereas the UMTS adopts root-raised-cosine pulses. **18**Finally, it needs observation times up to 50–100 symbols, which may be too long with fast fading. **19**The UMTS recommends estimation times within ten symbols.

20In this letter, we propose a channel estimator that is compatible with the UMTS recommendations and operates in an iterative fashion according to the space-alternating generalized expectation– maximization (SAGE) algorithm [10]. **21**At each step the parameters of a given path are estimated and the signal contribution of that path is canceled out from the received signal. **22**Compared with the ML estimator, it is much simpler to implement as it reduces a maximization problem with many parameters to a succession of simple one-dimensional searches. **23**The proposed scheme can be used with *any* training sequence and *any* pulse shape and provides estimates within five to eight symbols. **24**As in [8], MAI is treated as colored Gaussian noise and the inverse of its covariance matrix is used as a whitening filter to mitigate the near–far effect. **25**In computing the MAI covariance matrix, we assume that the interfering users have already been acquired and their channel parameters are known at the receiver [7]. **26**Robustness to errors in measuring these parameters is investigated through simulation. **27**The rest of the letter is organized as follows. **28**Section II describes the signal model and formulates the estimation problem. **29**In Section III, we address the estimation of path gains and delays for a single user. **30**Simulation results are discussed in Section IV and some conclusions are drawn in Section V.