UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE PATENT TRIAL AND APPEAL BOARD

_____

TOYOTA MOTOR CORP. and
CONTINENTAL AUTOMOTIVE SYSTEMS, INC.,
Petitioner,

v.

INTELLECTUAL VENTURES II LLC,
Patent Owner.

_____

IPR2022-00972
Patent 9,681,466 B2

_____

Before KRISTEN L. DROESCH, DAVID C. McKONE,
and AMBER L. HAGY, *Administrative Patent Judges*.

McKONE, *Administrative Patent Judge*.

DECISION

Granting Institution of *Inter Partes* Review
35 U.S.C. § 314

I. INTRODUCTION

Toyota Motor Corp. and Continental Automotive Systems, Inc., (collectively, "Petitioner") filed a Petition (Paper 1, "Pet.") requesting *inter partes* review of claims 1–10 of U.S. Patent No. 9,681,466 B2 (Ex. 1001, "the '466 patent"). Intellectual Ventures II LLC ("Patent Owner") filed a Preliminary Response to the Petition. Paper 7 ("Prelim. Resp."). Petitioner filed a Preliminary Reply to the Preliminary Response (Paper 12, "Prelim. Reply") and Patent Owner filed a Preliminary Sur-reply to the Preliminary Reply (Paper 14, "Prelim. Sur-reply").

We have authority to determine whether to institute an *inter partes* review. *See* 35 U.S.C. § 314 (2016); 37 C.F.R. § 42.4(a) (2019). The standard for instituting an inter partes review is set forth in 35 U.S.C. § 314(a), which provides that an *inter partes* review may not be instituted "unless . . . there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition." For the reasons explained below, we institute an *inter partes* review of the '466 patent.

II. BACKGROUND

A. Related Proceedings

The parties identify several district court and *inter partes* review proceedings in which the '466 patent is or has been involved. Pet. 72–74; Paper 4, 2.

B. The '466 Patent

The '466 patent relates to queuing algorithms for packet data transmission, for example, for use in the universal mobile telecommunication standard (UMTS). Ex. 1001, 1:23–26. Such algorithms can be used in a system such as illustrated in Figure 1, reproduced below:

FIG. 1

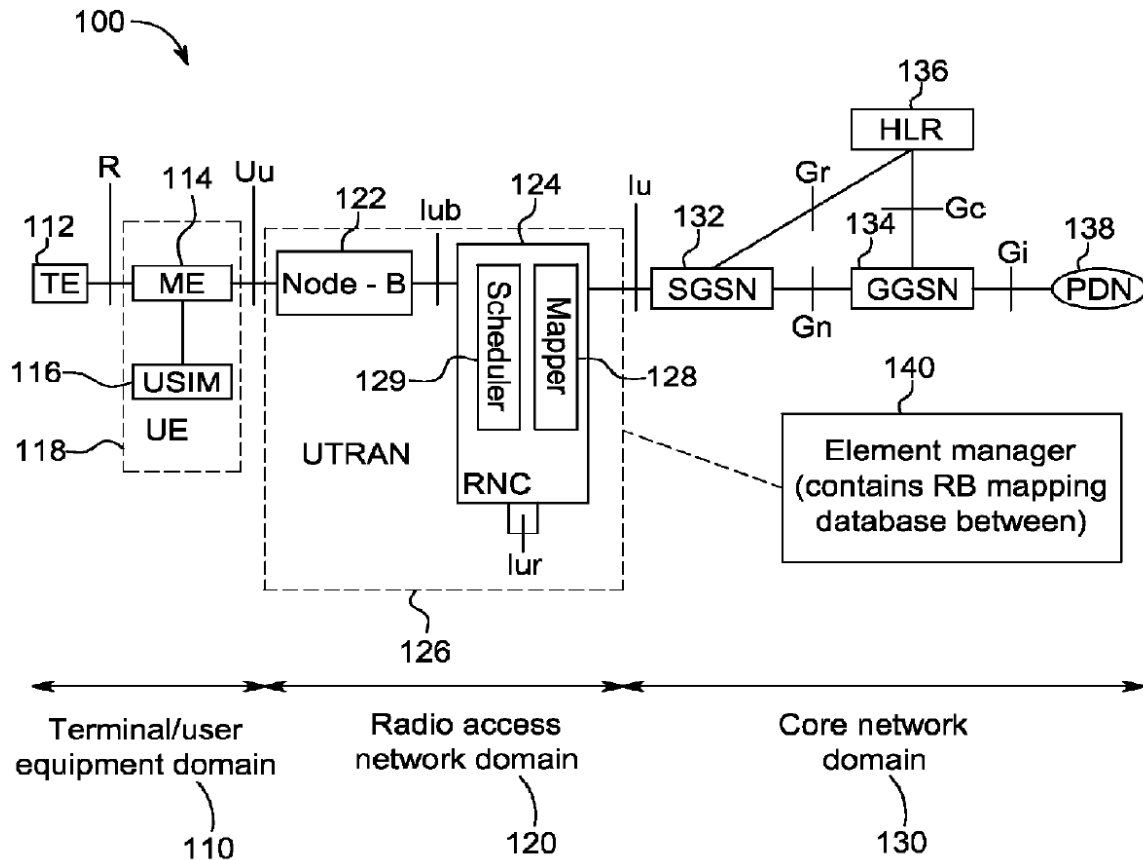Figure 1 is a block diagram of a 3rd Generation Partnership Project (3GPP) cellular communication system, e.g., UMTS Radio Access Network (UTRAN) system 100. *Id.* at 5:33–34, 6:8–10. UTRAN 100 includes terminal equipment (TE) 112 connected to mobile equipment (ME) 114, which, along with user service identity module (USIM) 116, constitutes user equipment (UE) 118. *Id.* at 6:13–18. UE 118 can be, for example, a mobile

station, a personal digital assistant, or a laptop computer communicating over an air interface using a cellular communication system. *Id.* at 6:18–23.

UE 118 communicates with base station (Node-B) 122 via wireless interface Uu. *Id.* at 6:24–26. Node-B 122 communicates with radio network controller (RNC) 124. *Id.* at 6:26–28. Mapper 128 of RNC 124 maps internet protocol (IP) packets to separate radio bearers (RBs). *Id.* at 7:40–44. Scheduler 129 allocates a certain proportion of a radio resource to each of the RBs. *Id.* at 7:45–47. Node-B 122 and RNC 124 together form UTRAN 126. *Id.* at 6:31–32. RNC 124 communicates with serving general packet radio service (GPRS) service node (SGSN) 132, which communicates with gateway GPRS support node (GGSN) 134, which, in turn, communicates with public data network (PDN) 138. *Id.* at 6:34–40.

The '466 patent describes a mechanism to provide prioritization of services across multiple users in an uplink (from UE 118 to Node-B 122) direction. *Id.* at 9:19–21. A single physical allocation of resources can be made to UE 118. *Id.* at 9:29–31. "This single physical allocation of radio resources provided to a UE is shared amongst a number of services, in proportion to their respective QoS requirements, for example in proportion to the signalled weight values (S_{tier})." *Id.* at 9:32–36. Scheduler logic in the network and UE "takes the single physical allocation and splits it between different RBs using the weight, S_{tier}, values." *Id.* at 9:46–50.

Figure 6, reproduced below, illustrates an example queuing algorithm:

600⟶

```
┌────────────────────────────────────┐
│  Determine normalised queue weighting │
│           parameter W' q             │
└────────────────────────────────────┘
                  │
605               ▼
┌────────────────────────────────────┐
│   Determine the number of RAAU allocated │
│              to each queue           │
└────────────────────────────────────┘
615               │
                  ▼
  ◇ Is no.        ┌────────────────────────────────────┐
  of queues   610 │        Recalculate RAAUq            │
  having allocated│  620                                │
  resources > EM  └────────────────────────────────────┘
  parameter ?     │                  ▼
        625       ┌────────────────────────────────────┐
                  │    Order the queues in order of RAAUq' │
                  └────────────────────────────────────┘
        628       │                  ▼
                  ┌────────────────────────────────────┐
                  │  Select first max_number_queues_serviced │
                  └────────────────────────────────────┘
        630       │                  ▼
                  ┌────────────────────────────────────┐
                  │            Set Nq' to zero           │
                  └────────────────────────────────────┘
        635       │                  ▼
                  ┌────────────────────────────────────┐
                  │  In all tiers selected above set Nq' to Nq │
                  └────────────────────────────────────┘
        640       │                  ▼
                  ┌────────────────────────────────────┐
                  │    Recalculate Wq using the Nq' vector │
                  └────────────────────────────────────┘
        645       │                  ▼
                  ┌────────────────────────────────────┐
                  │   Determine the modified number of RAAU │
                  │           allocated to each queue    │
                  └────────────────────────────────────┘
        650       │                  ▼
                  ┌────────────────────────────────────┐
                  │  Determine whether the sum of all RAAU │
                  │   allocated is greater than free RAAU │
                  └────────────────────────────────────┘
        655       │                  ▼
                  ┌────────────────────────────────────┐
                  │           Modify the RAAUq           │
                  └────────────────────────────────────┘
        660       │                  ▼
                  ┌────────────────────────────────────┐
                  │  Set RAAUq" for one of the queues (there may │
                  │  be multiple queues below the minimum) which │
                  │   has RAAUq" less than the minimum to zero │
                  └────────────────────────────────────┘
        665       │                  ▼
                  ┌────────────────────────────────────┐
                  │  Allocate these spare resources first to any other │
                  │  queues which are below the minimum. If there are │
                  │  still spare resources then allocate in turn 1 RAAU │
                  │   each to the queues that currently have │
                  │           RAAUq"> zero               │
                  └────────────────────────────────────┘
        670       │                  ▼
                  ┌────────────────────────────────────┐
                  │           Update RAAUq               │
                  └────────────────────────────────────┘
        675       │                  ▼
                  ┌────────────────────────────────────┐
                  │  Update the running difference between the actually │
                  │   allocated RAAU (RAAUq" ) and the originally │
                  │      determined allocated RAAU RAAUq:  │
                  └────────────────────────────────────┘
```
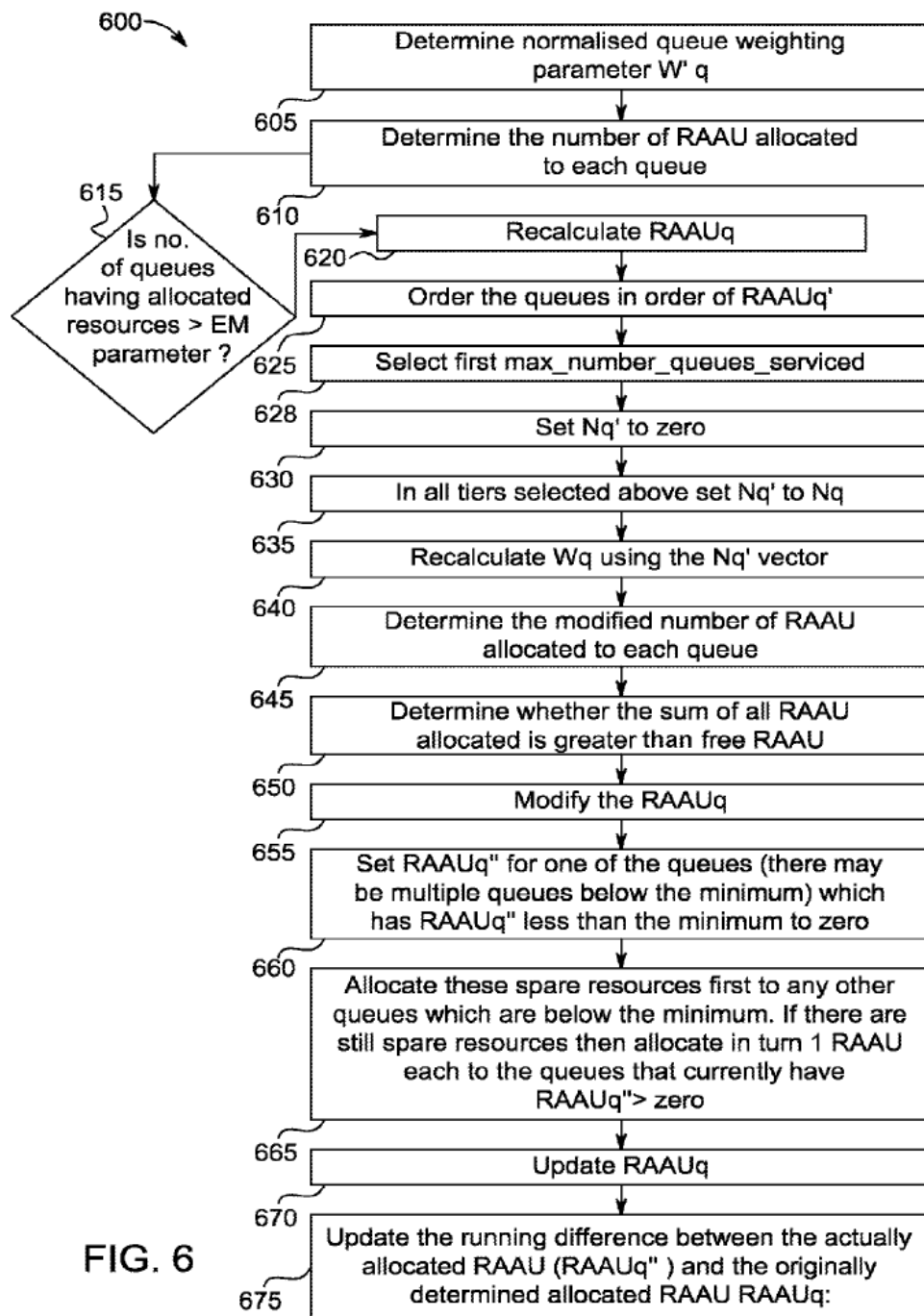
FIG. 6

Figure 6 is a flow chart of a method to limit a number of services/queues to be allocated RBs at a particular instant in time. *Id.* at 5:47–49. "The algorithm may be run when a number of active users (i.e. those users known to have a buffer occupancy greater than zero in any queue) is greater than a

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

fastcase®
Smarter legal research.