Source: InterDigital Communications, LLC
Title: Uplink power control procedures and Text Proposal for E-UTRA
Agenda Item: 6.4.2
Document for: Discussion and decision

1. Introduction

This contribution describes UE and eNodeB behavior for uplink power control. We first present a description of the procedures, incorporating agreements reached at #50bis [2], [3], and then give text modifications for TR36.213 [1].

2. General procedures for uplink power control

2.1 Physical uplink shared channel

The PUSCH power control has two components, an open loop and a closed-loop component. Both open and closed loop components run consecutively, but asynchronously. The procedure can be viewed as the following flow diagram:
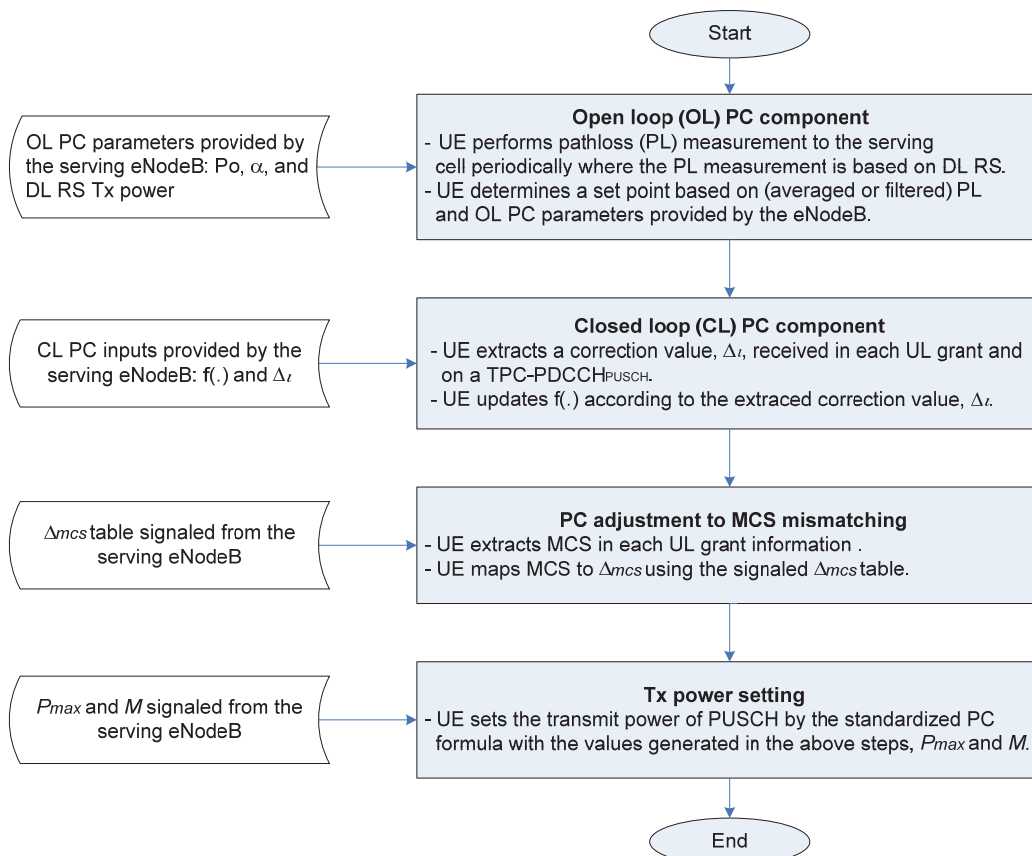

Figure 1 Flow chart for PUSCH power control

In a given uplink subframe i, PUSCH may be transmitted

- on a dynamically assigned resource (by an uplink grant on PDCCH)
- or on a persistently assigned resource
- or not at all

In a given downlink subframe n, a power control command $\Delta_{PUSCH}(n)$ for PUSCH power control may be provided

- within an UL scheduling grant on PDCCH
- or on a TPC-PDCCH (referred to as the TPC-PDCCH$_{PUSCH}$)
- or not at all

If, in a given downlink subframe n, power control commands are provided within an UL scheduling grant *and* on a TPC-PDCCH$_{PUSCH}$, the valid power control assumed to be delivered in subframe n should be the power control command provided within the UL scheduling grant (i.e. the power control command on TPC-PDCCH$_{PUSCH}$ should be ignored).

If PUSCH is transmitted in subframe i, it is transmitted with the power P_{PUSCH} *(i)*. In the power control formula below, K_{PUSCH} is the delay in the PUSCH power control, i.e. a power control command provided in downlink subframe n will not impact the PUSCH transmit power in subframes prior to subframe $n+K_{PUSCH}$. Exact value of K_{PUSCH} is TBD.

The power control formula for PUSCH, as modified in [3], is outlined below:

$$P_{PUSCH}(i) = \min(P_{\max}, 10\log_{10}(M) + P_o + \alpha \cdot PL + \Delta_{mcs} + f[\Delta_{PUSCH}(i - K_{PUSCH})])$$

where:

- $P_{\max}$ is the maximum allowed power (in dBm) that depends on the UE power class

- M is the number of assigned resource blocks as indicated in the UL scheduling grant

- P_o is a UE specific parameter (in dBm) with 1 dB resolution over the range: [-126dBm, 24dBm]

- α is cell specific path loss compensation factor (can be set to one to allow full path loss compensation) that has 8 values from 0.4 to 1 in steps of 0.1 with one of the possible values being zero.

- PL is the downlink pathloss calculated in the UE from a RSRP measurement and signaled RS transmit power

- Δ_{mcs} is signaled by RRC (Δ_{mcs} table entries can be set to zero)

  o MCS signaled in each UL scheduling grant

- Δ_{PUSCH} is a UE specific correction value and is included in every N[th] UL scheduling grant, (where N can be 1), or jointly coded with other UE specific correction values on a TPC-PDCCH$_{PUSCH}$.

  o The UE attempts to detect a TPC-PDCCH$_{PUSCH}$ on every subframe except when in DRX.

The power control formula is applied differently dependent on scheduling as given by:

- Dynamically scheduled PUSCH
  - Function $f[*]$ represents either accumulation or absolute value, The mode is signaled semi-statically via higher layers. When a new value of Δ_{PUSCH} is received in the Nth scheduling grant,
    - For absolute control
$$f[\Delta_{PUSCH}(i\text{-}K_{PUSCH})] = \Delta_{PUSCH}(i\text{-}K_{PUSCH})$$
    - For accumulation control
$$f[\Delta_{PUSCH}(i-K_{PUSCH})] = \sum_{m=0}^{i}\{\Delta_{PUSCH}(m-K_{PUSCH})\}$$

- Persistently scheduled PUSCH
  - Function $f[*]$ represents accumulation only. When a new value of Δ_{PUSCH} is received in the TPC-PDCCH$_{PUSCH}$,
$$f[\Delta_{PUSCH}(i-K_{PUSCH})] = \sum_{m=0}^{i}\{\Delta_{PUSCH}(m-K_{PUSCH})\}$$

- Combined dynamically and persistently scheduled PUSCH
  - When UL grant is configured for accumulation commands (for dynamically scheduled PUSCH), the UE combines the accumulation commands received in both the SG and the TPC-PDCCH$_{PUSCH}$
  - When UL grant is configured for absolute commands (for dynamically scheduled PUSCH), the UE resets the accumulation immediately after receiving each absolute TPC and then combines the absolute with the accumulation

$$f[\Delta_{PUSCH}(i-K_{PUSCH})] = \Delta_{PUSCH,UL\_grant}(j-K_{PUSCH}) + \sum_{m=j+1}^{i}\{\Delta_{PUSCH,TPC-PDCCH}(m-K_{PUSCH})\}$$

  where j is the subframe index of the last absolute command.

If no power control step $\Delta_{PUSCH}(k)$ is provided on neither an UL scheduling grant or on a TPC-PDCCH$_{PUSCH}$ in subframe k, then

- For absolute control, $\Delta_{PUSCH}(k)$ in the equations above should be set to the latest value of Δ_{PUSCH}.

- For accumulation control, $\Delta_{PUSCH}(k)$ in the equations above should be set to zero.

2.1.1 Open loop component

The UE first determines the open loop component based on a filtered linear pathloss estimate, *pl,* from the serving eNodeB to the UE. The pathloss is updated in the power control formula after each downlink RSRP measurement. The UE continuously (or periodically) measures the instantaneous pathloss at the k-th instance based on the DL RSRP whose transmit RS power is signaled to the UE.

A filtering method is then applied to the pathloss measurements, such as

$$pl_k = \rho \cdot pl_{k-1} + (1-\rho) \cdot pl_k^{inst}$$

where pl_k and pl_{k-1} represents the filtered pathloss at the k-th instance and (k-1)-th instant, respectively. pl_k^{inst} is the instantaneous pathloss at the k-th instant. ρ is a filter coefficient, $0 \leq \rho \leq 1$, which is generally determined by the UE, possibly depending on pathloss variation, fast fading rate, the time of UL transmission, etc. Then, the pathloss in dB is obtained as

$$PL_k = 10log_{10}(pl_k).$$

Alternatively, a moving averaging method may be considered for the pathloss filtering.

2.1.2 Closed loop component

Additionally, the UE applies a closed-loop power correction factor relative to the open loop power, primarily in order to compensate for open loop errors, including the pathloss estimation error due to non-perfect reciprocity in UL and DL in FDD, and the UE impairments due to power amplifier and receiver non-linearities.

The correction factor is derived at the UE, based on the recent received correction command(s). Which UL scheduling grants convey the correction command is a higher layer configurable parameter per UE basis, so that the UE knows which UL grants to look at for the correction command. For example, the command signaling is done in particular UL scheduling grants such as the UL grant associated with a pre-defined HARQ process, say, HARQ process #1; not every grant corresponding to that HARQ process has to carry a TPC command. In this case, it is assumed that multiple control channel formats (such as one with a PC correction command and another without it) are supported in the downlink. No additional control signaling to indicate a control channel format in use in a given TTI is required, since the UE knows in advance which UL grants carry the correction command. We also assume the UE applies BD (blind detection) to determine which grant format was sent. This BD can be based on message length, CRC mask, code or other methods.

BD protects against the problem of a lost TPC command if a SG (scheduled grant) is not detected. If the kth grant containing $\Delta_{PUSCH}(k)$ is lost, the UE would not transmit on the assigned allocation due to the grant failure. The eNodeB would, of course, not receive the expected transmission. The eNodeB may then append Δ_{PUSCH} to the next scheduling grant. Upon successful detection the UE, having not received the previous grant, would consider this one the kth grant.

Another scenario is when the UE correctly receives the kth grant with $\Delta_{PUSCH}(k)$, resets its counter, transmits its message, but the eNodeB fails to correctly receive the UE's transmission. Following the logic above, the eNodeB may resend $\Delta_{PUSCH}(k)$ in the next grant. However, the UE knows to ignore this redundant power control command.

For scheduled UEs there are two alternatives for applying the correction $f[\Delta_{PUSCH}]$ to the Tx power, accumulated or absolute value. The eNodeB sends to each scheduled UE (or a sub-group of scheduled UEs) a power correction factor using multiple command bits, 2 bits for accumulation control or TBD [2 or 3] bits for absolute control, in the UL grant (or on the TPC-PDCCH$_{PUSCH}$ for accumulation), where the correction command is determined based on link quality (such as received PSD or SINR) of the UL power controlled data channel (and possibly UL sounding reference symbol, if available). For instance, assuming absolute control, the correction factor may be determined such as

$$\Delta_{PUSCH} = \lfloor SINR_{Target} - ESINR_{est} \rfloor$$

where $ESINR_{est}$ and $SINR_{Target}$ denote the effective SINR (ESINR) estimate at the receiver and target SINR, respectively, of the power controlled channel(s) in dB. [x] denotes a correction value in the correction set which is nearest to x. The observed samples at the eNodeB for the ESINR estimation include (some of or all) SC-FDMA symbols of the UL power controlled channel(s), which have been received over an averaging window. Because slow power control is to be used, a linear block average since the last correction command signaling may be used when commands are sent infrequently, or a moving average over a suitable window may be used if updates are sent in every grant.

For persistenlty scheduled UEs only accumulation control is used. For accumulated control the eNodeB sends a command in every N grants or in the TPC-PDCCH$_{PUSCH}$. Assuming block averaging is used, the command is derived as follows:

For each reception from the UE, the eNodeB computes an error given by

$$e_k = SINR_{Target} - ESINR_k$$

where $ESINR_k$ is the effective SINR, in dB, for the kth reception since the last command was sent. The eNodeB computes a block average over M receptions during N intervals:

$$E = (\sum_1^M 10^{0.1e_k})/M$$

The command sent is obtained as

$$\Delta_{PUSCH} = \left[10\log_{10} E\right]$$

where [x] denotes a correction value in the correction set which is nearest to x. If the command is sent frequently, e.g. every grant, then a moving average over a window M can be used.

2.1.3 Timing of closed loop corrections

To reduce the signaling overhead of the power correction command, the correction command signaling is not required in every UL grant. Assuming that there exist multiple DL control formats, we can reduce the signaling overhead by applying the following rules:

- A correction command signaling timing is configured at the eNodeB (or on a RRC level) per UE basis and is then known at both the eNodeB and the UE via higher layer signaling.

- When the correction command is signaled in the UL grant, assuming that UL HARQ is synchronous, the signaling timing configuration can be simplified such that the command signaling is done in particular UL grants such as the UL grant associated with a pre-defined HARQ process, say, HARQ process #1. However, even in this case it is not necessary to signal the correction commands in all the associated UL grant channels. For example, the signaling may occur in every N associated grant channel for N >= 1, which would be equivalent to one command signaling in every N HARQ cycle period. Figure 2Figure 2 shows an example of the proposed PC scheme when the PC correction command is conveyed in the UL grant associated with HARQ process #1 and N is set to 2. In this example, the PC update rate is 8 msec, assuming the number of HARQ processes is 4 and the inter-TTI is equal to 1.

- The signaling timing (or associated parameters) may be reconfigured on a semi static basis.

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.