Adaptive Games for Agile Spectrum Access Based on Extended Kalman Filtering

Duo Zhang, *Member, IEEE*, and Zhi Tian, *Senior Member, IEEE*

***Abstract*—For dynamic spectrum allocation (DSA), distributed game has emerged as an attractive approach that enhances radio spectrum utilization efficiency at scalable complexity in network size. Cognitive radios act as game players to judiciously decide their transmission power spectrum density (TPSD) based on channel state information (CSI), which is generally estimated independently from the DSA games and may cause large computational and communication overheads. Particularly in the presence of doubly selective fading channels, a conventional DSA game needs to re-train the channel estimator and re-calculate the TPSD decisions for every transmission burst. To enable DSA intelligence at affordable costs, this paper proposes novel adaptive DSA algorithms based on channel tracking. Under the framework of extended Kalman filter (EKF), both the unknown CSI and the TPSD are modeled into a state vector that is to be tracked dynamically. This approach leads to EKF-based adaptive games that jointly track the CSI and update TPSD decisions, resulting in fast convergence and reduced communication overhead. Simulations are performed to testify the effectiveness of the proposed DSA algorithms, in terms of the achieved system spectrum efficiency, communication overhead, as well as resilience to user mobility.**

***Index Terms*—Channel tracking, distributed game, doubly selective fading, dynamic spectrum allocation, extended Kalman filter.**

I. Introduction

In THE emerging wireless access paradigm of dynamic spectrum allocation (DSA), spectrum-agile cognitive radios (CRs) opportunistically gain access to temporarily unused frequency bands in order to improve the network spectral efficiency [1], [2]. A popular approach to DSA is via distributed games, which yield efficient radio resource allocation at scalable complexity in the network size [2]. CRs are self-interested game players, each of which optimizes its local utility function by taking action from the action space defined by available spectrum and allowable transmission power. DSA boils down to determining the transmission power spectral density (TPSD) of each CR for efficient spectrum utilization.

Recent research on the game approach to DSA has focused on the game-theoretic and information-theoretic aspects, which reveal the achievable rates on spectral efficiency, and enhancement strategies for designing local utility functions [3]–[5]. These works rely on perfect channel state information (CSI), and largely ignore the communication overhead incurred by dynamic gaming. In practice, the distributed nature of game-based DSA gives rise to a number of implementation challenges in terms of communication burdens, computational complexity, and the need for fast channel acquisition. Game players make individual TPSD decisions, broadcast messages (e.g., TPSD decisions and/or interference prices [4]) via a common control channel, and may iterate the process through multiple rounds until convergence. This process causes a heavy communication burden on the control channel, which has been recognized as a bottleneck in implementing DSA. This problem is particularly aggravated in high-rate mobile systems experiencing doubly selective fading channels. DSA iterations have to converge within the time limitation of an invariant fading block [5]. For practical DSA schemes, it is essential to alleviate the communication burden over the control channel.

Agile spectrum access relies on spectrum sensing to obtain knowledge of the wireless environments. A CR needs to constantly monitor the dynamics of fading channels, while making decisions on its TPSD instantaneously. Constrained by limited power, the CR must balance its power usage between that for channel sensing and for data transmission, and thus cannot afford much computational complexity on game-oriented CSI estimation in practice. Furthermore, the CSI estimation errors may impact the game outcome [6], which has to be properly accounted for in order to identify practical transmission policies and channel estimation techniques that retain the benefits of DSA.

To cope with the aforementioned physical-layer challenges in DSA, this paper develops adaptive DSA schemes that account for the communication needs of dynamic sensing and gaming at practical complexity. We consider a distributed CR network in the presence of doubly selective fading channels. To expedite the convergence speed of DSA games in such unknown channels, we resort to Kalman filtering (KF) to adaptively track the channel dynamics and then incorporate the predicted CSI into the decision making process of DSA games. The KF techniques have been extensively studied for channel prediction and estimation of frequency selective fading channels, working well in both time domain (TD) [7] and frequency domain (FD) [8]. Tailoring to the need of CR applications, our goal here is not just to estimate and track the channel information, but to ultimately reach distributed TPSD decisions that adapt to the varying fading channels. To this end, we propose an extended Kalman filter (EKF) approach that incorporates the TPSD into

Manuscript received September 5, 2006; revised February 8, 2007. Z. Tian was supported in part by U.S. NSF Grant CCR-0238174. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Fulvio Gini.

The authors are with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI 49931-1295 USA (e-mail: duozhang@mtu.edu; ztian@mtu.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JSTSP.2007.897050

its state vector. The state-space equation with respect to the TPSD is derived from optimal water-filling power allocation in the frequency domain [5], which is nonlinear in nature. Two low-complexity DSA algorithms are then designed: an EKF-assisted game (EKFG) and an EKF-updated game (EKFUG). Both DSA algorithms make adaptive TPSD decisions based on new channel updates and past decisions, which result in fast convergence and thus save communication and computational overhead.

The rest of the paper is organized as follows. Section II outlines the system model and provides a primitive on game-based DSA under known CSI. Section III derives the state-space equation and measurement equation for the joint TPSD decision and channel tracking problem. A set of adaptive DSA algorithms are proposed subsequently, including EKFG, EKFUG, as well as some variants such as frequency-domain KF game (FDKFG). Section IV provides simulation results to compare the proposed DSA algorithms with both the ideal game (IG) under perfect CSI and a non-adaptive game based on linear MMSE (LMMSEG) channel estimation [6]. Concluding remarks follow in Section V.

Notation: $(\cdot)^*, (\cdot)^T, (\cdot)^H$ and $(\cdot)^\dagger$ denote conjugate, transpose, Hermitian transpose, and matrix pseudo-inverse, respectively. $E\{\cdot\}$ stands for expectation. $\mathcal{CN}(\boldsymbol{\mu}; \boldsymbol{\Upsilon})$ denotes complex Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Upsilon}$. $\mathbf{I}_M$ denotes the identity matrix of size M; $\mathbf{0}_{M\times N}$ and $\mathbf{1}_{M\times N}$ denote an $M \times N$ all-zero and all-one matrix, respectively. $\mathrm{vec}(\mathbf{x}_1, \ldots, \mathbf{x}_M)$ denotes $[\mathbf{x}_1^T, \ldots, \mathbf{x}_M^{T\,T}]$; $[\cdot]_p$ denotes the pth entry of a vector; $[\mathbf{x}]_{p\ldots q}$ denotes a vector composed by pth to qth entries of a vector $\mathbf{x}$; and $[\cdot]_{p,q}$ denotes the (p,q)th entry of a matrix. $(\mathbf{X})^+$ denotes a matrix whose (p,q)th entry is $\max\{[\mathbf{X}]_{p,q}, 0\}$; The symbol $\otimes$ denotes Kronecker product; $\mathbf{X} \odot \mathbf{Y}$ and $\mathbf{X} \oslash \mathbf{Y}$ denote element-by-element multiplication and division, respectively.

II. Modeling and Problem Statement

A. Multi-CR System in Doubly-Selective Fading Channels

Consider a network of spectrum-agile users sharing access to a total of B_{tot} Hz in the frequency range $[f_0^b, f_{N_f}^b]$. Depending on the frequency locations of active legacy users, the radio spectrum is segmented into N_f consecutive yet non-overlapping frequency bands. The kth band is located at $[f_k^b, f_{k+1}^b]$, with center frequency $f_k = (f_{k+1}^b + f_k^b)/2$ and bandwidth $B_k = f_{k+1}^b - f_k^b$, $k = 0, \ldots, N_f - 1$.

Through medium access control, active CR users may register through a control channel and perform DSA within each session during which the user distribution remains unchanged. Suppose that there are N_u users in the session, and the session time is much longer than the channel coherence time. Each cognitive user corresponds to one pair of unicast transmitting and receiving nodes. For notational convenience, the mth CR user, the jth transmitter, and the mth receiver are denoted as CR_m, Tx_j and Rx_m, respectively. Focusing on the process of DSA, this sequel adopts the following assumptions.

- The channel delay spread is bounded by $\tau_{\max}$, while the maximum Doppler frequency is $f_d^{(jm)}$ on the link between Tx_j and Rx_m.
- Without loss of generality, the N_f frequency bands are partitioned such that the PSD of the channel response is smooth and almost flat within each band, i.e., B_k is no greater than the coherence bandwidth $1/\tau_{\max}$, $\forall k$. The bandwidths $\{B_k\}_{k=0}^{N_f-1}$ are not necessarily equal.
- The frequency boundaries $\{f_k^b\}_{k=0}^{N_f}$ are known to the CRs, either as *a priori* knowledge or having been estimated via spectral detection and classification techniques [2], [8].
- There is no central spectrum controller in the network, mandating distributed DSA. Each cognitive receiver Rx_m estimates the received signal PSD, decides on the TPSD, and feeds the decisions back to its dedicated transmitter Tx_m via a perfect feedback channel.

To account for high-rate transmissions and user mobility, the propagating channels are modeled as doubly selective fading channels. Corresponding to all the $(\mathrm{Tx}_j, \mathrm{Rx}_m)$ pairs, there are a total of N_u^2 wireless links, of which N_u are dedicated data channels and the rest are interference channels [2]. Reflecting frequency selectivity, the channel impulse response between Tx_j and Rx_m is expressed by [8], [12]

$$\breve{h}^{(jm)}(t;\tau) = \sum_{l=0}^{L-1} \breve{h}_{t,l}^{(jm)} \delta(\tau - lT) \tag{1}$$

where $T = 1/B_{\mathrm{tot}}$ and L is an upper bound of the multipath number (e.g., $\tau_{\max} \leq (L-1)T$). The channel taps $\breve{h}_{t,l}^{(jm)}$, $\forall l$, are typically assumed to be low-pass zero-mean circular complex Gaussian processes.

We adopt the commonly-used block fading model in which the channel response is treated as invariant within each block. We limit our treatment on time selectivity to block-by-block slow fading. The block size is given by $N_b = \lfloor T_b/T \rfloor$, where T_b is the channel coherence time. Let $h^{(jm)}(t;f)$ denote the Fourier transform (FT) of $\breve{h}^{(jm)}(t;\tau)$ in (1) with respect to τ. Sampling at $t = nT_b$ $(n = 1, 2, \ldots)$ and $f = f_k$ $(k = 0, \ldots, N_f - 1)$, the discrete frequency-domain (FD) channel response of the $(\mathrm{Tx}_j, \mathrm{Rx}_m)$ link on the kth band at the nth time block is given by

$$h_n^{(jm)}(k) := h^{(jm)}(nT_b; f_k) = \mathbf{f}_k^T \mathbf{h}_n^{(jm)} \tag{2}$$

where $\mathbf{f}_k := [1, e^{-j2\pi T f_k}, \ldots, e^{-j2\pi T f_k (L-1)}]^T$ and $\mathbf{h}_n^{(jm)} = [\breve{h}_{nT_b,0}^{(jm)}, \ldots, \breve{h}_{nT_b L-1}^{(jm)}]^T$. Defining $\mathbf{F} := [\mathbf{f}_0, \ldots, \mathbf{f}_{N_f-1}]^T$ and $\boldsymbol{\mathcal{H}}_n^{(jm)} := [h_n^{(jm)}(0), \ldots, h_n^{(jm)}(N_f - 1)]^T$, we express the channel transfer function at the nth block as

$$\boldsymbol{\mathcal{H}}_n^{(jm)} = \mathbf{F}\mathbf{h}_n^{(jm)}. \tag{3}$$

In the presence of mobility-induced Doppler shifts, the channel dynamics can be modeled by an auto-regressive (AR) process [10]. A pth-order AR model on $\mathbf{h}_n^{(jm)}$ is given by [12]

$$\mathbf{h}_n^{(jm)} = -\sum_{i=1}^{p} \mathbf{A}_i^{(jm)} \mathbf{H}_{n-i}^{(jm)} + \mathbf{Q}^{(jm)} \mathbf{u}_{T,n}^{(jm)} \tag{4}$$

where $\mathbf{u}_{T,n}^{(jm)}$ is an $L \times 1$ complex Gaussian noise vector obeying $\mathcal{CN}(\mathbf{0}_{L\times 1}, \mathbf{I}_L)$. In the time domain (TD), documented experiments suggest that the AR parameters $\mathbf{A}_i^{(jm)}$ and $\mathbf{Q}^{(jm)}$, can be

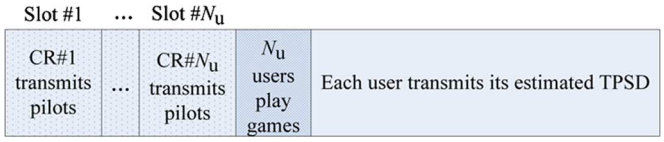

Fig. 1. Transmission scheme of an N_u-CR system: nth block.

obtained by solving a standard Yule-Walker equation based on the correlation matrices $\mathbf{D}_{\mathbf{h}}^{(jm)}(\cdot)$ defined by [10]

$$\left[\mathbf{D}_{\mathbf{h}}^{(jm)}(n-n')\right]_{l,l'} := E\left\{\breve{h}_{nT_b l}^{(jm)}\left(\breve{h}_{n'T_b l'}^{(jm)}\right)^*\right\} = \xi_{l,l'}^{(jm)}\mathbf{A}_{n-n'}^{(jm)}, \quad (5)$$

where $\xi_{l,l'}^{(jm)}$ is the correlation factor of the tap gain, $\mathbf{A}_{n-n'}^{(jm)} = J_0(2\pi(n-n')f_d^{(jm)}T_b)$, and $J_0(\cdot)$ is the zeroth-order Bessel function of the first kind. The TD AR models in (4) and (5) form the basis for accurately describing the physical channels in the frequency domain.

We collect the TD channel taps over every p consecutive blocks into a state vector $\mathbf{x}_{T,n}^{(jm)} := [(\mathbf{h}_n^{(jm)})^T, \ldots, (\mathbf{h}_{n-p+1}^{(jm)})^T]^T$. Following (4), $\mathbf{x}_{T,n}^{(jm)}$ obeys a state-space equation in the form

$$\mathbf{x}_{T,n}^{(jm)} = \mathbf{\Phi}_T^{(jm)}\mathbf{x}_{T,n-1}^{(jm)} + \mathbf{\Lambda}_T^{(jm)}\mathbf{u}_{T,n}^{(jm)} \quad (6)$$

where $\mathbf{\Phi}_T^{(jm)} = \begin{pmatrix} -\mathbf{Z}_p^{(jm)} & -\mathbf{A}_p^{(jm)} \\ \mathbf{I}_{L(p-1)} & \mathbf{0}_{(p-1)L\times L} \end{pmatrix}$, $\mathbf{Z}_p^{(jm)} := [\mathbf{A}_1^{(jm)}, \ldots, \mathbf{A}_{p-1}^{(jm)}]$ and $\mathbf{\Lambda}_T^{(jm)} = [(\mathbf{Q}^{(jm)})^T, \mathbf{0}_{L\times(p-1)L}]^T$.

B. Transmission Scheme

In the distributed game approach to DSA, each CR makes decisions on its own transmission power spectrum density (TPSD) based on the CSI of all the N_u channels sensed at its receiver. To acquire such CSI, we adopt a pilot-based transmission scheme depicted in Fig. 1. During each session, active CRs coarsely synchronize through a control channel and start block transmission. At the beginning of each block, CRs take turns to transmit pilots one user at a time, while all other transmitters are silent. After all N_u users have transmitted, each of the N_u receivers will have acquired the CSI on all received channels. Multi-player gaming is performed to determine the preferred TPSD of each CR, which will be used in the ensuing information data transmission within this block.

Suppose that each user transmits M' pilot symbols per block, followed by $L-1$ zeros padded at the tail of the pilots to guard against inter-user interference caused by multipath. It is thus required that $MN_u \leq N_b$, where $M = M' + L - 1$. Let us collect the pilot symbols of Tx_j at the nth block by an $M' \times 1$ vector $\mathbf{p}_n^{(j)} := [p_n^{(j)}(0), \ldots, p_n^{(j)}(M'-1)]^T$. The ith pilot symbol received at Rx_m from Tx_j is given by

$$y_n^{(jm)}(i) = \sum_{l=0}^{L-1}\sqrt{E_p^{(j)}}\breve{h}_{nT_b,l}^{(jm)}p_n^{(j)}(iT-lT) + w_n^{(jm)}(i) \quad (7)$$

where $E_p^{(j)}$ is the pilot transmission power and $w_n^{(jm)}(i)$ is the observation noise. Accordingly, the received sample vector $\mathbf{y}_{T,n}^{(jm)} := [y_n^{(jm)}(M-1), \ldots, y_n^{(jm)}(0)]^T$ is related to the unknown channel state vector $\mathbf{x}_{T,n}^{(jm)}$ by

$$\mathbf{y}_{T,n}^{(jm)} = \mathbf{\Psi}_T^{(j)}\mathbf{x}_{T,n}^{(jm)} + \mathbf{w}_{T,n}^{(jm)} \quad (8)$$

where $\mathbf{\Psi}_T^{(j)} = \sqrt{E_p^{(j)}}[[[\breve{\mathbf{p}}_n^{(j)}]_{M+L-1\cdots M}, \ldots, [\breve{\mathbf{p}}_n^{(j)}]_{L\cdots 1}]^T, \mathbf{0}_{M\times(p-1)L}]$ is the $M \times pL$ transmission matrix built on the zero-prefixed vector $\breve{\mathbf{p}}_n^{(j)} := [\mathbf{0}_{1\times(L-1)}, (\mathbf{p}_n^{(j)})^T, \mathbf{0}_{1\times(L-1)}]^T$, and $\mathbf{w}_{T,n}^{(jm)} := [w_n^{(jm)}(M-1), \ldots, w_n^{(jm)}(0)]^T$ is the complex Gaussian noise obeying $\mathcal{CN}(\mathbf{0}, \mathbf{\Omega}_T^{(jm)})$.

C. Preliminaries on Game-Based DSA and Problem Statement

For each CR_m, the TPSD distribution can be specified by a vector $\mathbf{s}_n^{(m)} := [s_n^{(m)}(0), \ldots, s_n^{(m)}(N_f-1)]^T$, where $s_n^{(m)}(k)$ denotes the TPSD allocated to the kth band at the nth block. Let $\mathbf{B} := [B_0, \ldots, B_{N_f-1}]^T$ and $P^{(m)}$ denote the total transmit power of CR_m. Under a game framework, the action space of CR_m is defined by allowable $\mathbf{s}_n^{(m)}$ subject to the power constraint $\mathbf{B}^T\mathbf{s}_n^{(m)} = P^{(m)}$. When the FD CSI is known, each CR_m locally maximizes the utility defined by its information-theoretic data rate over all N_f bands [2]

$$\max_{\mathbf{s}_n^{(m)}} \sum_{k=0}^{N_f-1} B_k \times \log\left(1 + \frac{s_n^{(m)}(k)|h_n^{(mm)}(k)|^2\,\Gamma^{-1}}{n^{(m)}(k) + \sum_{j\neq m} s_n^{(j)}(k)|h_n^{(jm)}(k)|^2}\right) \quad (9)$$

where $n^{(m)}(k)$ is the ambient noise spectrum and the SNR gap Γ is a scalar constant [2]. Each CR makes its own decision on $\mathbf{s}_n^{(m)}$ based on its own objective, but implicitly interacts with other users via $\{\mathbf{s}_n^{(j)}\}_{j\neq m}$, until reaching a Nash Equilibrium, if existent [5].

There are a variety of game types, among which we consider games of complete information. This means that all the parameters (power constraints, channel gains, etc.) are common knowledge to all users in the system [3]. In practice, some of these parameters have to be measured, and the corresponding measurements must be exchanged among the CRs. In a conventional one-shot game, each CR chooses its power allocation once and for all, which incurs minimal communication overhead, but may not attain high sum utility. In an iterative game, CRs take turns to make individual TPSD decisions via (9) once per user in each iteration of the game, broadcast messages/decisions via a common control channel, and repeat multiple iterations until convergence. DSA via iterative games enhances the system capacity over one-shot games. However, it causes a heavy communication burden on the control channel.

The goal of this paper is to derive optimal solutions to acquiring the CSI estimates needed for DSA among mutiple CRs. In a general setup, each CR does not know the CSI a priori, but just knows the measurements $\mathbf{y}_{T,n}^{(jm)}$ that relate to TD CSI via (8). Particularly when an iterative game is adopted for DSA in

the presence of slowly time-varying channels, we develop adaptive means for joint CSI tracking and TPSD updating via extended Kalman filtering. The proposed techniques save computational load via adaptive processing, and result in fast convergence via joint updating of both CSI and TPSD, which in turn reduces the communication overhead incurred during the transient phase of iterative games.

To facilitate the CSI estimation and TPSD decision making process, we also suggest a general transmission structure depicted in Fig. 1. Our focus here is to identify an enabling transmission scheme for practical multi-CR games in a general manner, rather than to advocate a specific pilot pattern or a transmission format for some specific system architecture such as CDMA or OFDM. Nevertheless, for a specific system (including CDMA and OFDM), once the corresponding pilot-pattern matrix $\mathbf{\Psi}_T^{(j)}$ and the observation equation (8) are formulated, our proposed DSA algorithms directly apply.

There are some other forms of games such as simultaneous games and repeated games, which may entail different gaming processes and communication loads. Nevertheless, the channel tracking scheme developed in this paper is directly applicable to other games, whenever CSI becomes relevant as in (9).

III. EKF-Based Adaptive DSA Games

This section presents an extended Kalman filter (EKF) approach that jointly handles channel tracking and adaptive TPSD decisions. Frequency-domain state equation and measurement equation are derived to form the basis of EKF games. A decomposition strategy is introduced to reduce the computational load of the EKF. Based on the decomposed EKF, two low-complexity game-based adaptive DSA algorithms are presented, and their computational complexities are quantified.

A. Channel Tracking via Frequency-Domain Kalman Filter

To track time-varying fading channels, time domain Kalman filter (TDKF) has been widely used. The TD state-space (6) and measurement (8) together define a TDKF engine for tracking the state vector $\mathbf{x}_{T,n}^{(jm)}$ containing TD CSI. Since our goal here is to determine the TPSD in the frequency domain, we want to develop an equivalent FDKF for estimating FD CSI. To this end, we define an FD state vector $\mathbf{x}_{F,n}^{(jm)} := [(\boldsymbol{\mathcal{H}}_n^{(jm)})^T, \ldots, (\mathcal{H}_{n-p+1}^{(jm)})^T]^T$. Note from (3) that $\boldsymbol{\mathcal{H}}_n^{(jm)}$ is nothing but the FT of $\mathbf{h}_n^{(jm)}$. Thus, it holds that $\mathbf{x}_{F,n}^{(jm)} = (\mathbf{I}_p \otimes \mathbf{F})\mathbf{x}_{T,n}^{(jm)}$ and $\mathbf{x}_{T,n}^{(jm)} = (\mathbf{I}_p \otimes \mathbf{F}^\dagger)\mathbf{x}_{F,n}^{(jm)}$. Defining $\mathbf{\Psi}_F^{(j)} = \mathbf{\Psi}_T^{(j)}(\mathbf{I}_p \otimes \mathbf{F}^\dagger)$, the FD measurement equation arises from (8) as

$$\mathbf{y}_{T,n}^{(jm)} = \mathbf{\Psi}_F^{(j)}\mathbf{x}_{F,n}^{(jm)} + \mathbf{w}_{T,n}^{(jm)}. \tag{10}$$

Multiplying $(\mathbf{I}_p \otimes \mathbf{F})$ on both sides of (6), we obtain the FD state-space equation as follows:

$$\mathbf{x}_{F,n}^{(jm)} = \mathbf{\Phi}_F^{(jm)}\mathbf{x}_{F,n-1}^{(jm)} + \mathbf{\Lambda}_F^{(jm)}\mathbf{u}_{T,n}^{(jm)} \tag{11}$$

where $\mathbf{\Phi}_F^{(jm)} := (\mathbf{I}_p \otimes \mathbf{F})\mathbf{\Phi}_T^{(jm)}(\mathbf{I}_p \otimes \mathbf{F}^\dagger)$ and $\mathbf{\Lambda}_F^{(jm)} := (\mathbf{I}_p \otimes \mathbf{F})\mathbf{\Lambda}_T^{(jm)}$.

With (10) and (11) in place, an FD tracker for updating the FD CSI $\mathbf{x}_{F,n}^{(jm)}$ can be derived from the standard KF. Let $\hat{\mathbf{\Sigma}}_{F,n\,|\,n-1}^{(jm)}$ and $\hat{\mathbf{\Sigma}}_{F,n\,|\,n}^{(jm)}$ denote the predicted and updated covariance matrices of the predicted and updated FD CSI vectors $\hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}$ and $\hat{\mathbf{x}}_{F,n\,|\,n}^{(jm)}$ respectively, and $\mathbf{K}_{F,n}^{(jm)}$ denote the Kalman gain. The following FDKF procedure arises.

FDKF Algorithm for Estimating FD CSI

$$\begin{aligned}
\text{Predict}: \hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)} &= \mathbf{\Phi}_F^{(jm)}\hat{\mathbf{x}}_{F,n-1\,|\,n-1}^{(jm)} \\
\hat{\mathbf{\Sigma}}_{F,n\,|\,n-1}^{(jm)} &= \mathbf{\Phi}_F^{(jm)}\hat{\mathbf{\Sigma}}_{F,n-1\,|\,n-1}^{(jm)}\left(\mathbf{\Phi}_F^{(jm)}\right)^H \\
&\quad + \mathbf{\Lambda}_F^{(jm)}\left(\mathbf{\Lambda}_F^{(jm)}\right)^H \\
\text{Update}: \quad \mathbf{\Pi}_{F,n}^{(jm)} &= \mathbf{\Psi}_F^{(j)}\hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}\left(\mathbf{\Psi}_F^{(j)}\right)^H + \mathbf{\Omega}_T^{(m)} \\
\mathbf{K}_{F,n}^{(jm)} &= \hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}\left(\mathbf{\Psi}_F^{(j)}\right)^H\left(\mathbf{\Pi}_{F,n}^{(jm)}\right)^{-1} \\
\hat{\mathbf{x}}_{F,n\,|\,n}^{(jm)} &= \hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)} \\
&\quad + \mathbf{K}_{F,n}^{(jm)}\left(\mathbf{y}_{T,n}^{(jm)} - \mathbf{\Psi}_F^{(j)}\hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}\right) \\
\hat{\mathbf{\Sigma}}_{F,n\,|\,n}^{(jm)} &= \left(\mathbf{I}_{pN_f} - \mathbf{K}_{F,n}^{(jm)}\mathbf{\Psi}_F^{(j)}\right)\hat{\mathbf{\Sigma}}_{F,n\,|\,n-1}^{(jm)}
\end{aligned}$$

Apparently, another approach to track the FD CSI $\mathbf{x}_{F,n}^{(jm)}$ is to first perform TDKF for updating $\mathbf{x}_{T,n}^{(jm)}$, followed by applying FT in the form of $\mathbf{x}_{F,n}^{(jm)} = (\mathbf{I}_p \otimes \mathbf{F})\mathbf{x}_{T,n}^{(jm)}$. It can be shown that these two approaches are equivalent, while the TDKF approach has computational advantages over FDKF. Nevertheless, the intermediate steps in FDKF are useful in constructing ensuing joint TPSD and channel tracking schemes that operate in the frequency domain.

B. TPSD Tracking via Extended Kalman Filter

Tailoring to DSA needs, we next expand the FDKF structure to incorporate the TPSD vector $\mathbf{s}_n^{(m)}$ into the state vector. By doing so, CRs will be able to make TPSD decisions adaptively by tracking the expanded state vector via KF. As we will show later, the state equation for $\mathbf{s}_n^{(m)}$ turns out to be nonlinear in FD CSI, which motivates EKF. Since EKF involves state-vector derivatives defined only for real-valued variables, we would partition the channel state vector into real and imaginary parts. Adopting the operator $\tilde{\mathbf{x}} := [\mathrm{Re}\{\mathbf{x}\}^T, \mathrm{Im}\{\mathbf{x}\}^T]^T$, we define for each CR_m an expanded *joint* state vector $\mathbf{x}_{E,n}^{(m)} := [(\mathbf{s}_n^{(m)})^T, (\mathbf{x}_{F,n}^{(m)})^T]^T$, where $\mathbf{x}_{F,n}^{(m)} := [(\tilde{\mathbf{x}}_{F,n}^{(1m)})^T, \ldots, (\tilde{\mathbf{x}}_{F,n}^{(N_u m)})^T]^T$ collects all the FD CSI vectors related to Rx_m. Similarly, we collect all the measurements received by Rx_m into a composite vector $\mathbf{y}_{E,n}^{(m)} := [(\tilde{\mathbf{y}}_{T,n}^{(1m)})^T, \ldots, (\tilde{\mathbf{y}}_{T,n}^{(N_u m)})^T]^T$. In mathematical abstraction, we denote the mapping from state $\mathbf{x}_{E,n-1}^{(m)}$ to state $\mathbf{x}_{E,n}^{(m)}$ and the mapping from $\mathbf{x}_{E,n}^{(m)}$ to measurements $\mathbf{y}_{E,n}^{(m)}$ as

$$\begin{aligned}
\mathbf{x}_{E,n}^{(m)} &= \phi(\mathbf{x}_{E,n-1}^{(m)}, \mathbf{U}_{E,n}^{(m)}), \\
\mathbf{y}_{E,n}^{(m)} &= \psi(\mathbf{x}_{E,n}^{(m)}, \mathbf{w}_{E,n}^{(m)})
\end{aligned} \tag{12}$$

respectively, where $\mathbf{U}_{E,n}^{(m)} := [(\tilde{\mathbf{u}}_{T,n}^{(1m)})^T, \ldots, (\tilde{\mathbf{u}}_{T,n}^{(N_u m)})^T]^T$ and $\mathbf{w}_{E,n}^{(m)} := [(\tilde{\mathbf{w}}_{T,n}^{(1m)})^T, \ldots, (\tilde{\mathbf{w}}_{T,n}^{(N_u m)})^T]^T$.

The closed-form expressions for the mapping functions $\phi(\cdot)$ and $\psi(\cdot)$ in (12) will be presented in Section III.C. Assuming they are known and noting the nonlinear nature of $\phi(\cdot)$, an EKF can be directly applied to not only update all the FD CSI estimates $\hat{\mathbf{x}}_{F,n}^{(m)}$ tracked by Rx_m, but also predict the TPSD $\hat{\mathbf{s}}_n^{(m)}$ to be used by Tx_m. Let $\mathbf{\Upsilon}_E^{(m)}$ ($\mathbf{\Upsilon}_E^{(m)} = (1/2)\mathbf{I}_{2N_uL}$) and $\mathbf{\Omega}_E^{(m)}$ denote the covariance matrices of the stacked noise vectors $\mathbf{U}_{E,n}^{(m)}$ and $\mathbf{w}_{E,n}^{(m)}$, respectively. A TPSD tracker is given by the following standard EKF steps.

EKF Algorithm for Jointly Tracking TPSD and CSI

$$\text{Predict}: \hat{\mathbf{x}}_{E,n\,|\,n-1}^{(m)} = \phi\left(\hat{\mathbf{x}}_{E,n-1\,|\,n-1}^{(m)}, \mathbf{u}_{E,n}^{(m)}\right)\Big|_{\mathbf{u}_{E,n}^{(m)}=\mathbf{0}} \tag{13a}$$

$$\mathbf{\Phi}_{E,n}^{(m)} = \partial\phi/\partial\hat{\mathbf{x}}_E^{(m)}\Big|_{(\hat{\mathbf{x}}_{E,n-1\,|\,n-1}^{(m)},\mathbf{0})} \tag{13b}$$

$$\mathbf{\Lambda}_{E,n}^{(m)} = \partial\phi/\partial\mathbf{U}_{E,n}^{(m)}\Big|_{(\hat{\mathbf{x}}_{E,n-1\,|\,n-1}^{(m)},\mathbf{0})} \tag{13c}$$

$$\hat{\mathbf{\Sigma}}_{E,n\,|\,n-1}^{(m)} = \mathbf{\Phi}_{E,n}^{(m)}\hat{\mathbf{\Sigma}}_{E,n-1\,|\,n-1}^{(m)}(\mathbf{\Phi}_{E,n}^{(m)})^H + \mathbf{\Lambda}_{E,n}^{(m)}\mathbf{\Upsilon}_E^{(m)}(\mathbf{\Lambda}_{E,n}^{(m)})^H \tag{13d}$$

$$\text{Update}: \mathbf{\Psi}_{E,n}^{(m)} = \partial\psi/\partial\hat{\mathbf{x}}_E^{(m)}\Big|_{(\hat{\mathbf{x}}_{E,n\,|\,n-1}^{(m)},\mathbf{0})} \tag{13e}$$

$$\mathbf{\Pi}_{E,n}^{(m)} = \mathbf{\Psi}_{E,n}^{(m)}\hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}(\mathbf{\Psi}_{E,n}^{(m)})^H + \mathbf{\Omega}_E^{(m)} \tag{13f}$$

$$\mathbf{K}_{E,n}^{(m)} = \hat{\mathbf{x}}_{F,n\,|\,n-1}^{(jm)}(\mathbf{\Psi}_{E,n}^{(m)})^H(\mathbf{\Pi}_{E,n}^{(m)})^{-1} \tag{13g}$$

$$\hat{\mathbf{x}}_{E,n\,|\,n}^{(m)} = \hat{\mathbf{x}}_{E,n\,|\,n-1}^{(m)} + \mathbf{K}_{E,n}^{(m)}\mathbf{y}_{E,n}^{(m)} - \mathbf{K}_{E,n}^{(m)}\psi\left(\mathbf{\Psi}_{\mathbf{E,n}}^{(\mathbf{m})}\hat{\mathbf{x}}_{\mathbf{E,n\,|\,n-1}}^{(\mathbf{m})}, \mathbf{w}_{\mathbf{E,n}}^{(\mathbf{m})}\right)\Big|_{\mathbf{w}_{E,n}^{(m)}=\mathbf{0}} \tag{13h}$$

$$= \hat{\mathbf{x}}_{E,n\,|\,n-1}^{(m)} + \mathbf{K}_{E,n}^{(m)}\Big(\mathbf{y}_{E,n}^{(m)} - \psi \cdot \left(\mathbf{\Psi}_{\mathbf{E,n}}^{(\mathbf{m})}\hat{\mathbf{x}}_{\mathbf{E,n\,|\,n-1}}^{(\mathbf{m})}\mathbf{w}_{\mathbf{E,n}}^{(\mathbf{m})}\right)\Big|_{\mathbf{w}_{\mathbf{E,n}}^{(\mathbf{m})}=\mathbf{0}}$$

$$\hat{\mathbf{\Sigma}}_{E,n\,|\,n}^{(m)} = (\mathbf{I}_{(2pN_u+1)N_f} - \mathbf{K}_{E,n}^{(m)}\mathbf{\Psi}_{E,n}^{(m)})\hat{\mathbf{\Sigma}}_{E,n\,|\,n-1}^{(m)} \tag{13i}$$

C. Adaptive DSA Algorithms

The EKF algorithm in (13a)–(13i) offers a framework for adaptively updating the DSA decisions on the TPSD vector $\hat{\mathbf{s}}_n^{(m)}$. The remaining issues are to define the mapping functions $\phi(\cdot)$ and $\psi(\cdot)$, and to solve the gradient steps in (13a)–(13i).

From Section III-A, we know that the CSI-related state-vector element $\mathbf{x}_{F,n}^{(m)}$ and the measurements $\mathbf{y}_{E,n}^{(m)}$ are independent of $\mathbf{s}_n^{(m)}$; hence, the corresponding state-space functions mapping from $\mathbf{x}_{F,n-1}^{(m)}$ to $\mathbf{x}_{F,n}^{(m)}$ and the measurement function mapping from $\mathbf{x}_{E,n}^{(m)}$ to $\mathbf{y}_{E,n}^{(m)}$ can be derived straightforwardly from (10) and (11), both being linear. To define the mapping function for the remaining state-vector element $\mathbf{s}_n^{(m)}$, we re-visit the utility function in (9). Evidently, the optimal solution to $\mathbf{s}_n^{(m)}$ is determined by all the FD CSI related to Rx_m as well as the TPSD vectors of all other users. Let $\chi(\cdot)$ denote the function expressing the optimal $\mathbf{s}_n^{(m)}$ in (9), which is given by water-filling [5]

$$\begin{aligned}\mathbf{s}_n^{(m)} &= \chi\left(\mathcal{H}_n^{(1m)},\ldots,\mathcal{H}_n^{(N_um)};\mathbf{s}_n^{(1)},\ldots,\mathbf{s}_n^{(m-1)},\mathbf{s}_n^{(m+1)},\ldots,\mathbf{s}_n^{(N_u)}\right)\\ &= \left(\mu_n^{(m)}\mathbf{1}_{N_f\times 1} - \left(\sum_{j\neq m}|\,\mathcal{H}_n^{(jm)}\,|^2\odot\mathbf{s}_n^{(j)} + \mathbf{n}^{(m)}\right)\oslash|\,\mathcal{H}_n^{(mm)}\,|^2\right)^+\end{aligned} \tag{14}$$

where $\mathbf{n}^{(m)} := [n^{(m)}(0),\ldots,n^{(m)}(N_f-1)]^T$, $\mathbf{B}^T\mathbf{s}_n^{(m)} = \tilde{P}^{(m)}$, $\tilde{P}^{(m)} := \Gamma^{-1}P^{(m)}$ is the effective power limit, and $\mu_n^{(m)}$ is a scalar loading factor chosen to satisfy the power constraint. The operator $(\cdot)^+$ ensures that the TPSD allocation does not assign negative power density to any bands. When all the elements in $\mathbf{s}_n^{(m)}$ are positive, that is, there is no zero point in $\mathbf{s}_n^{(m)}$, CR_m will transmit power over all N_u bands.

Our adaptive DSA problem boils down to the following remaining question: *how to solve the unknown predicted joint state* $\hat{\mathbf{x}}_{E,n\,|\,n-1}^{(m)}$, *the state-related matrices* $\mathbf{\Phi}_{E,n}^{(m)}$ *and* $\mathbf{\Lambda}_{E,n}^{(m)}$, *and the measurement-related matrices* $\mathbf{\Psi}_{E,n}^{(m)}$, *using the water-filling solution so that the EKF Algorithm can track the TPSD? In another words, how to solve* (13a)–(13c), (13e) *based on* (14)?

Appendix A answers this question and derives the detailed steps in the EKF (13a)–(13i). Accordingly, we design two game-based DSA algorithms: EKF-assisted game (EKFG) and EKF-updated game (EKFUG). Each algorithm involves a DSA game that starts from CR_1 and goes through all CRs without loss of generality. The following *Lemma 1* is useful in developing these algorithms (proof omitted for space limit).

Lemma 1: If there is no zero point in the updated TPSD $(\hat{\mathbf{s}}_{n\,|\,n}^{(m)})^+$, then the power constraint $\mathbf{B}^T\hat{\mathbf{s}}_{n\,|\,n}^{(m)} = \tilde{P}^{(m)}$ is guaranteed automatically.

Our proposed EKFG and EKFUG algorithms using *Lemma 1* are described by the following steps.

Step 1) At the beginning of the nth block, set $m=1$ to let CR_1 start.

Step 2) CR_m employs the EKF in (13a)–(13i) to update the TPSD $\hat{\mathbf{s}}_{n\,|\,n}^{(m)}$. In the prediction step (13a), the predicted TPSD $\hat{\mathbf{s}}_{n\,|\,n-1}^{(m)}$ is computed from the $\chi(\cdot)$ function in (14) using $\hat{\mathbf{s}}_n^{(1)},\ldots,\hat{\mathbf{s}}_n^{(m-1)}\mathbf{s}_{n-1}^{(m+1)},\ldots,\mathbf{s}_{n-1}^{(N_u)}$. Here, $\{\hat{\mathbf{s}}_n^{(j)}\}_{j=1}^{m-1}$ are the updated TPSD vectors of the nth block, while the rest (N_u-m) terms $\{\mathbf{s}_{n-1}^{(j)}\}_{j=m+1}^{N_u}$ are the previous TPSD from the $(n-1)$th block.

Step 3) If there is no zero point in the TPSD $(\hat{\mathbf{s}}_{n\,|\,n}^{(m)})^+$, the power constraint $\mathbf{B}^T\hat{\mathbf{s}}_{n\,|\,n}^{(m)} = \tilde{P}^{(m)}$ is satisfied automatically, which is guaranteed by *Lemma 1*. When zero points exist in $(\hat{\mathbf{s}}_{n\,|\,n}^{(m)})^+$, for power constraint guarantee, a small water-level adjustment $\Delta\mu$ is imposed by solving a standard water-filling equation

$\mathbf{B}^T(\hat{\mathbf{s}}_{n\,|\,n}^{(m)}+\Delta\mu)^+ = \tilde{P}^{(m)}$ and then setting $\hat{\mathbf{s}}_{n\,|\,n}^{(m)} = (\hat{\mathbf{s}}_{n\,|\,n}^{(m)}+\Delta\mu)^+$. The CR_m broadcasts its updated TPSD on the control channel.

Step 4) Set $m = m+1$. If $m \le N_u$, go back to Step 2). Otherwise, one game round has completed; for EKFUG, go to Step 6); for EKFG, go to Step 5).

Step 5) After the first-round EKF, the N_u users resume to perform an iterative game and go through multiple iterations. During iterations, CRs take turns to compute the TPSD from the standard water-filling response in (14), using the FD CSI estimated from the first-round iteration completed in Step 4). After convergence, go to Step 6).

Step 6) Each CR uses the estimated TPSD for transmitting information data until the nth block ends. Set $n = n+1$ and go to Step 1) to start tracking and transmission in the next block.

The difference between the EKFG and EKFUG procedures lies in the additional Step 5) that EKFG performs. An iterative game in Step 5) offers the EKFG with better spectral efficiency at the expense of higher communication overhead, as we will demonstrate via simulations in Section IV.

Some remarks are due on the communication overhead incurred over the control channel. Corresponding to Fig. 1, during the training phase when N_u users take turns to transmit pilots, they need to send signaling messages over the control channel to indicate the conclusion of their pilot transmission so that the next CR can start transmitting. During the gaming phase, in Steps 4) and 5), each user needs to broadcasts its tentative TPSD decision (and interference price when applicable) over the control channel. Such overhead is inherent in sequential games and iterative games with perfect information, in exchange for better achieved utility than one-shot games that do not involve information exchanges.

D. Efficient Implementation of EKFG and EKFUG

The EKF offers a natural framework for designing adaptive games. On the other hand, it incurs considerable computational complexity because of the matrix inversion. The joint state vector $\mathbf{x}_{E,n}^{(m)}$ is of length $N_f(2N_u+1)$, which can be large when N_u is large. To reduce the complexity of EKF, it is desired to decompose the joint state vector into smaller segments and track each segment via a small-size EKF in a parallel manner. *Lemma 2* demonstrate the feasibility of such a decomposition.

Lemma 2: Under the game framework, the EKF in Section III.B can be decomposed into an equivalent architecture consisting of a small-scale EKF for tracking the TPSD (EKF-TPSD) via (15a)–(15e), N_u parallel FDKFs for tracking the FD CSI, N_u complex-to-real KF mapping (C/R-KFM) modules and a composition KF mapping (KFM) module.

Proof: See Appendix B for the proof.

Let $\hat{\mathbf{\Sigma}}_{\zeta,n\,|\,n}^{(jm)}$ denote the updated covariance matrix of $\mathbf{x}_{F,n}^{(m)}$ (which stacks all the Rx_m-related FD CSI) at the nth block, $\mathbf{\Pi}_{\zeta,n}^{(m)}$ denote the residual covariance matrix of $\mathbf{x}_{F,n}^{(m)}$, and $\mathbf{K}_{S,n}^{(m)}$ denote the Kalman gain for updating *only* the TPSD. The EKF-TPSD algorithm is given in (15) [see Appendix B for details].

In the EKF-TPSD algorithm, each CR first makes a predicted game-decision on the TPSD $\hat{\mathbf{s}}_{n\,|\,n-1}^{(m)}$ as in (15a). Essentially, CR_m makes a predicted action in response to other CRs based on the state transition equations of all CR_m-tracked FD CSI. Then, the predicted TPSD is updated by Kalman gain and measurement residual in (15e) to yield the final TPSD. This is a step that CR_m modifies its predicted action based on measurements of all CR_m-tracked FD CSI. Meanwhile, the FD CSI of each of N_u users is tracked by a small-scale FDKF as shown in Appendix B. With such decomposition, the EKF steps can be implemented in the proposed adaptive DSA algorithms EKFG and EKFUG at much lower complexity.

EKF-TPSD Algorithm for Tracking TPSD

$$\text{Predict}: \quad \hat{\mathbf{s}}_{n\,|\,n-1}^{(m)} = \chi\left(\hat{\boldsymbol{\mathcal{H}}}_{n\,|\,n-1}^{(1m)},\ldots,\hat{\boldsymbol{\mathcal{H}}}_{n\,|\,n-1}^{(N_u m)};\hat{\mathbf{s}}_n^{(1)},\ldots,\hat{\mathbf{s}}_n^{(m-1)},\mathbf{s}_{n-1}^{(m+1)},\ldots,\mathbf{s}_{n-1}^{(N_u)}\right) \tag{15a}$$

$$\hat{\mathbf{\Sigma}}_{u,n\,|\,n-1}^{(m)} = \frac{\partial\chi}{\partial\mathbf{x}_{F,n-1}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{x}_{F,n-1}^{(m)}}\right)^H \tag{15b}$$

$$+\frac{\partial\chi}{\partial\mathbf{u}_{E,n}^{(m)}}\mathbf{\Upsilon}_E^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{u}_{E,n}^{(m)}}\right)^H$$

$$\text{Update}: \left(\mathbf{\Pi}_{E,n}^{(m)}\right)^{-1} = \left(\mathbf{\Pi}_{\zeta,n}^{(m)}\right)^{-1} \tag{15c}$$

$$\mathbf{K}_{S,n}^{(m)} = \hat{\mathbf{\Sigma}}_{u,n\,|\,n-1}^{(m)}\left(\frac{\partial\psi}{\partial\mathbf{x}_{F,n}^{(m)}}\right)^H\left(\mathbf{\Pi}_{E,n}^{(m)}\right)^{-1} \tag{15d}$$

$$\hat{\mathbf{s}}_{n\,|\,n}^{(m)} = \hat{\mathbf{s}}_{n\,|\,n-1}^{(m)} + \mathbf{K}_{S,n}^{(m)}\left(Y_{E,n}^{(m)} - \psi\left(\mathbf{\Psi}_{E,n}^{(m)}\mathbf{x}_{F,n\,|\,n-1}^{(m)},\mathbf{w}_{E,n}^{(m)}\right)\right)\Big|_{\mathbf{w}_{E,n}^{(m)}=\mathbf{0}}. \tag{15e}$$

So far, we have presented a couple of game-based DSA algorithms including EKFG and EKFUG, all implemented by the low-complexity EKF-TPSD algorithm. Furthermore, the FDKF algorithm for CSI tracking can also be expanded into a DSA game, which we term as FDKFG. In FDKFG, N_u small-scale FDKF is performed to track the decomposed CSI at the beginning of each block as in Section III-A, followed by a standard game as in (9) and (14) of Section II-C to compute the TPSD. The tracking procedures of all these KF-related adaptive DSA algorithms are summarized in Fig. 2, which shows the decomposition of EKF based on *Lemma 2*.

E. Comparison of Computational Complexity

It is of interest to compare the proposed adaptive DSA algorithms with reference to the non-adaptive DSA counterpart.

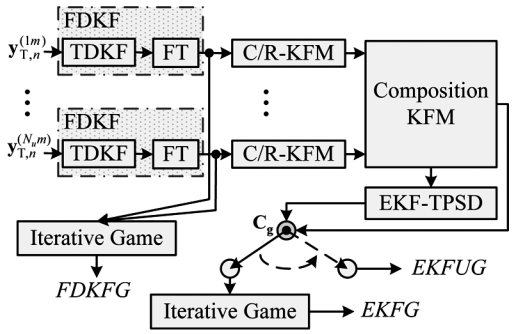

Fig. 2. Block diagrams of EKFG, EKFUG and FDKFG for CR_m.

TABLE I
COMPUTATIONAL COMPLEXITY PER CR

Algorithm	Matrix inversion	Real multiplications	Real additions
LMMSEG	$O(N_u M^3)$	$O(LN_u M^2)$	$O(LN_u M^2)$
FDKFG	$O(N_u M^3)$	$O(pLN_u M^2)$	$O(pLN_u M^2)$
EKFG&EKFUG	$O(N_u M^3)$	$O\big(p(N_f+L)N_u M^2\big)$	$O\big(p(N_f+L)N_u M^2\big)$

In [6], a cognitive receiver uses an LMMSE estimator to obtain the necessary TD CSI, which is then used in (9) to enable the standard game described in Section II-C. This procedure is performed block by block, without adaptation across blocks. We term this DSA algorithm as the LMMSE-assisted game (LMMSEG) and use it as a reference for comparison.

We first evaluate the comparative computational complexity. For Kalman-type filters, the filtering complexity is dominated by the matrix-inversion operation, compared to multiplication and addition operations. Suppose that a standard inversion technique is used which costs $\mathcal{O}(M^3)$ operations for an $M \times M$ matrix. The complexities of FDKFG, EKFG and EKFUG are listed in Table I, along with that of LMMSEG. It is shown that EKFG and EKFUG cost almost the same complexity as FDKFG and LMMSEG, since they all have the same complexity of matrix inversion. Compared with LMMSEG, the EKFG and EKFUG algorithms do incur a slight increase in complexity from their multiplications and additions, but the increase is trivial since a small p is sufficient for accurate AR modeling. It is worth emphasizing that the computational load of EKFG and EKFUG is made to be comparably small to that of LMMSEG through the decomposition steps that we introduce in *Lemma 2*. As such, the performance advantage of joint tracking of TPSD and CSI in our adaptive DSA comes at little complexity overhead compared with the separate processing in LMMSEG.

IV. SIMULATIONS

A. System Setup

Simulation examples are performed on a two-user CR network operating over the frequency range of $[901, 910]$ MHz, where $[901, 902]$ MHz is allocated to licensed mobile communication system as regulated by US FCC and $[902, 910]$ MHz belongs to license-free amateur radios. The $[902, 910]$ MHz range is divided into four bands each of 2 MHz bandwidth. Thus, $B_{\rm tot} = 9$MHz and $\mathbf{B} = [1, 2, 2, 2, 2]^T$ MHz. The sampling interval is $T = 1/B_{\rm tot} = 0.111$ μs.

In all simulations, $p = 1$ is employed, which has been shown to lead to effective tracking algorithms [10], [11]. The multipath channel adopted has two taps with un-correlated tap gains, i.e., $L = 2$ and $\xi_{l,l'} = \delta_{l,l'}$, which has been extensively used [13]. In the channel covariance matrix, the direct-channel gain is 3 dB stronger than the cross-channel gain.

The channel Doppler spread is caused by the two receivers moving at 30 km/h for CR_1 and 50 km/h for CR_2, which are typical speeds specified in ITU Veh-A. The channel coherence time is $T_b = 1$ ms to yield channel time-correlation $a_1^{(jm)} \in [0.98, 1)$, which falls within the permissible range for KF-based tracking to work. The block size is $N_b = T_b/T = 9000$, and the pilot arrangement is $M = 8$ [8].

The observation noise is assumed to be white, i.e., $\mathbf{\Omega}_T^{(jm)}$ is in the form of $(\sigma_w^{(jm)})^2 \mathbf{I}_M$. The measurement SNR of the pilots is given by $\mathrm{SNR}_o^{(m)} = E_p^{(m)} E\{\|\mathbf{h}_n^{(mm)}\|^2\}/(\sigma_w^{(jm)})^2$, and the data-transmission SNR is $\mathrm{SNR}_t^{(m)} = P^{(m)} E\{\|\mathbf{h}_n^{(mm)}\|^2\} \Gamma^{-1} (\sum_{k=0}^{N_f-1} n^{(m)}(k) B_k)^{-1}$. To initialize the EKF, $\hat{\mathbf{h}}_{0\,|\,0}^{(jm)}$ is generated randomly from $\mathcal{CN}(\mathbf{0}, \mathbf{D}_{\mathbf{H}}^{(jm)}(0))$, and $\hat{\mathbf{\Sigma}}_{T,0\,|\,0}^{(jm)} = \mathbf{D}_{\mathbf{H}}^{(jm)}(0)$, $\forall j, m \in [1, N_u]$. The KFs are trained for $N_{\rm tr} = 10$ blocks after initialization to avoid possible unstable states [8]. Simulation results after the $N_{\rm tr}$th block are collected to assess the performance metrics of interest, including the data rate loss, the communication burden and the TPSD estimation errors. Each test case generates 1000 channel realizations.

B. System Performance Versus SNR_o

The total system-rate of the game at the nth block is defined as the summation of all the CRs' data rates at the nth block. This quantity is indicative of the spectrum utilization efficiency, which is the motivating factor for DSA. The baseline for comparison is the total system-rate $R_n^{(\mathrm{IG})}$ of the ideal game (IG), which relies on the perfect CSI to make decisions on the TPSD. The system-rate of the EKFG is denoted by $R_n^{(\mathrm{EKFG})}$, while the system-rate loss is defined as $R_{\mathrm{loss},n}^{(\mathrm{EKFG})} = 1 - R_n^{(\mathrm{EKFG})}/R_n^{(\mathrm{IG})}$. The mean system-rate loss is defined as $\bar{R}_{\mathrm{loss}}^{(\mathrm{EKFG})} = E\{R_n^{(\mathrm{IG})} - R_n^{(\mathrm{EKFG})}\}/E\{R_n^{(\mathrm{IG})}\}$. At an outage probability $P_{\rm out}$, the outage system-rate loss is defined as $R_{\rm out}^{(\mathrm{EKFG})} = \arg_R\{Pr(R_{\mathrm{loss},n}^{(\mathrm{EKFG})} < R) = P_{\rm out}\}$. Similarly, $\bar{R}_{\mathrm{loss}}^{(\mathrm{LMMSEG})}$, $R_{\rm out}^{(\mathrm{LMMSEG})}$, $\bar{R}_{\mathrm{loss}}^{(\mathrm{EKFUG})}$ and $R_{\rm out}^{(\mathrm{EKFUG})}$ are defined for LMMSEG and EKFUG, respectively.

Fig. 3 depicts the mean and outage system-rate losses versus the observation SNR (SNR_o) for all the proposed DSA algorithms. For various $\mathrm{SNR}_{\rm o}$ values, the mean loss of EKFUG is better than LMMSEG and worse than EKFG. For example, EKFUG's mean system-rate loss over the ideal game is about 1.1%, falling between EKFG's 0.7% and LMMSEG's 1.9% at $\mathrm{SNR}_{\rm o} = 10$ dB. Its outage loss shows a similar trend, showing 2.6% at $\mathrm{SNR}_{\rm o} = 10$ dB while whose those of EKFG and LMMSEG are 2.1% and 3.7%, respectively. An interesting observation is that FDKFG performs worse than EKFUG, in terms of the outage system-rate loss. Their outage difference is small

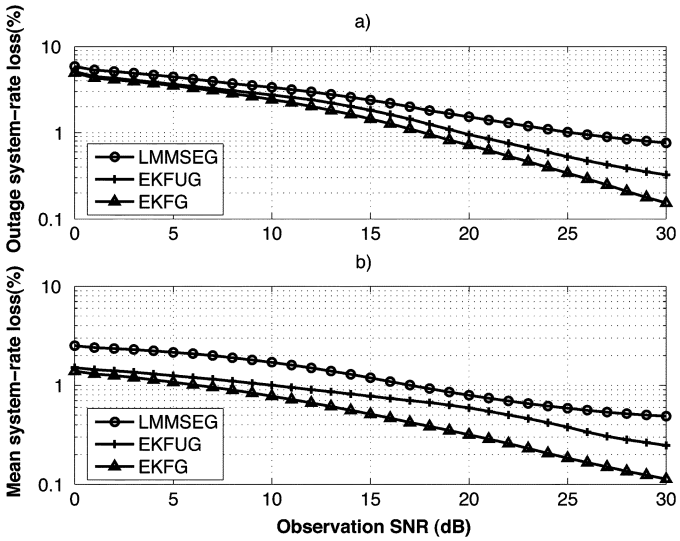

Fig. 3. Mean and outage system-rate losses versus observation SNR, $\mathrm{SNR}_t = 5$ dB, $v_1 = 30$ km/h, $v_2 = 50$ km/h.

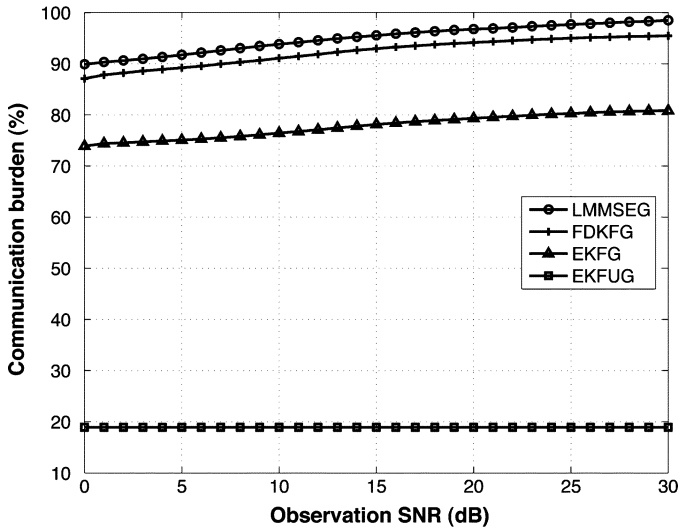

Fig. 4. Normalized communication burden versus observation SNR, $\mathrm{SNR}_t = 5$ dB, $v_1 = 30$ km/h, $v_2 = 50$ km/h.

at low SNRs and becomes large at high SNRs, and their mean difference shows a larger gap than their outage difference.

Another point of interest is the communication load of the proposed DSA algorithms. We suppose that the communication load for one user to announce its TPSD $\mathbf{s}_n^{(m)}$ once is Q bits. In an N_u-CR game, one *iteration* is defined as one round in which every CR takes action once in a sequential manner. Thus, if the game needs N_{iter} iterations to converge, the average communication burden of the game on the control channel is given by $(N_u E\{N_{\mathrm{iter}}\}Q)/(T_b)$ bits/s, since each user broadcasts its updated TPSD once per iteration on the control channel. Apparently, the communication burdens of these DSA games are reflected in the number of iterations performed.

Fig. 4 compares the communication burdens (on the control channel) of various algorithms normalized by that of IG. The number of iterations is determined by how fast the estimated TPSD approaches that of the IG, indicated by the system mean square error (MSE) defined for EKFG as

$$\mathrm{MSE}_{\mathrm{TPSD}}^{\mathrm{EKFG}} = \sum_{m=1}^{N_u} \frac{E\left\{\|\hat{\mathbf{s}}_n^{(\mathrm{EKFG},m)} - \mathbf{s}_n^{(\mathrm{IG},m)}\|\right\}}{E\left\{\|\mathbf{s}_n^{(\mathrm{IG},m)}\|\right\}}. \qquad (16)$$

The system MSEs of EKFUG and LMMSEG are similarly defined. An iterative game stops when the corresponding $\mathrm{MSE}_{\mathrm{TPSD}}$ is less than 10^{-3}. It is shown in Fig. 4 that the EKFUG algorithm has the lowest normalized burden at around 19% for various $\mathrm{SNR}_{\mathrm{o}}$s, whereas EKFG incurs a burden of about 75% and FDKF about 95% close to that of LMMSEG.

Overall, EFKUG has much less communication overhead, while its performance is quite close to that of IG and better than LMMSEG. Therefore, EFKUG is competitive for applications constrained by low-rate control channels. As for EKFG, it has the best system-rate performance while saving about 35% communication burden over the infeasible IG. Practically, EKFG attains the highest data-transmission rate, and hence the best spectrum efficiency, among all the DSA algorithms. Thus, EKFG is suitable for high data-rate applications that tolerate some moderate rate burden on the control channel.

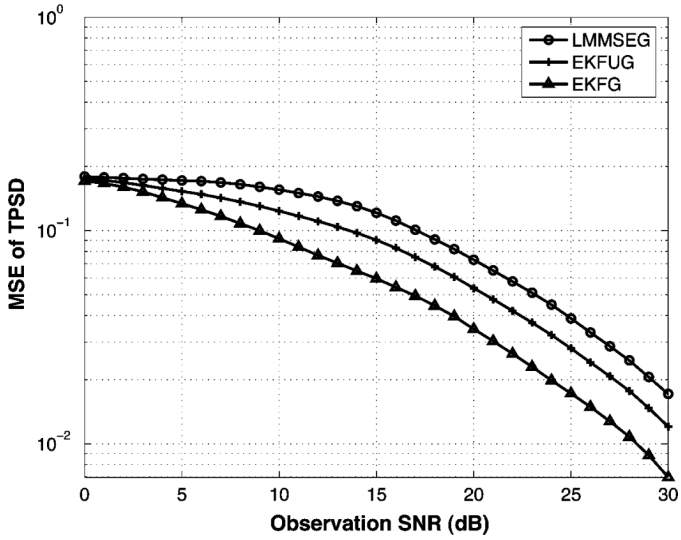

Fig. 5. Normalized MSE of TPSD versus observation SNR, $\mathrm{SNR}_t = 5$ dB, $v_1 = 30$ km/h, $v_2 = 50$ km/h.

Fig. 5 illustrates how close the proposed DSA schemes reach the ideal TPSD decisions. As in (16), the system MSE of a game is defined as the summation over the normalized MSEs of all the N_u TPSD decisions with reference to IG. As verified by Fig. 5, the greater the observation SNR is, the less the estimation error is, and thus the less the capacity loss of EKFG and EKFUG is. It can be observed that the system MSE performance of EKFUG lies in between that of LMMSEG and EKFG, consistent with the trend in their comparative system-rate loss performances.

The EKFUG algorithm makes decisions in only one iteration, while the rest games repeat multiple iterations until convergence. Therefore, the EKFUG is most economical in terms of saving communication burdens, compared with LMMSEG, FDKFG, and EKFG.

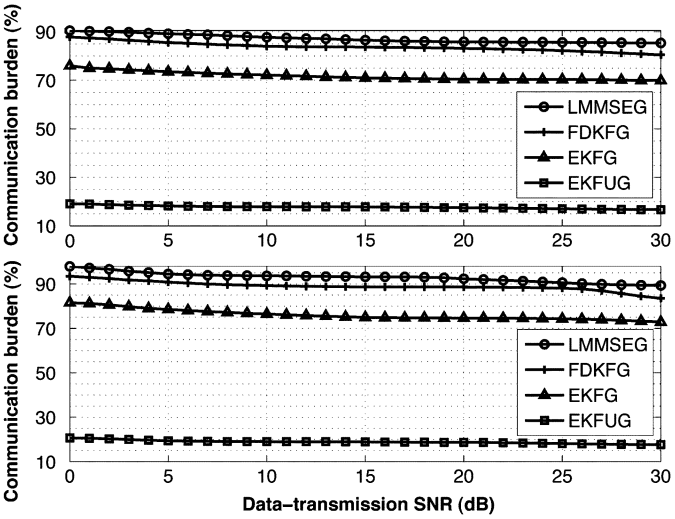

Fig. 6. Normalized communication burden versus data-transmission SNR, $\mathrm{SNR}_o = 5, 10$ dB, $v_1 = 30$ km/h, $v_2 = 50$ km/h.

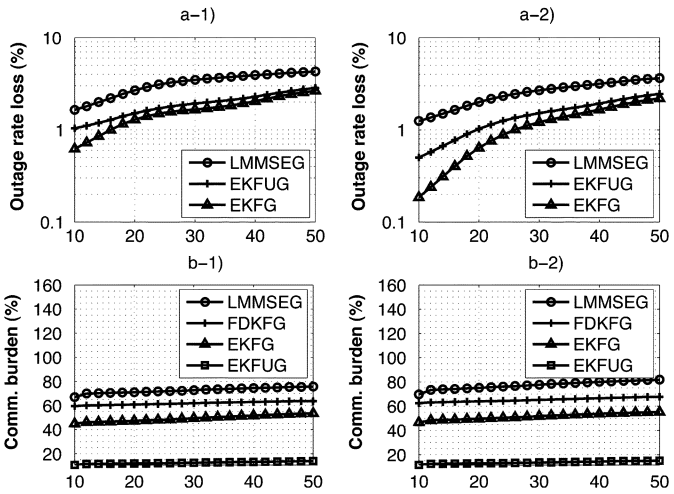

Fig. 7. Outage system-rate loss and communication burden versus CR speed, $\mathrm{SNR}_o = 5, 10$ dB, $\mathrm{SNR}_t = 5$ dB.

The proposed EKFG algorithm, being an iterative game, also entails less communication burden than LMMSEG and FDKFG. This is because it converges faster to the steady state, thanks to the KF-based channel tracking mechanism that memorizes history of past blocks. The FDKFG algorithm can be shown to converge to the same TPSD decisions as EFKG and thus yield the same data dates, but EKFG causes less communication burden than FDKFG by faster convergence.

C. Communication Burden Versus SNR_t

Fixing SNR_o used during the training phase, Fig. 6 depicts the communication burden versus data-transmission SNR (SNR_t). For the practical SNRs, EKFUG incurs the least communication burden while maintaining a moderately small system-rate loss over IG.

D. System Performance Versus Mobility

Fig. 7 evaluates the impact of CR speed on the performance. For convenience it is assumed that both CRs have the same speed varying from 10 km/h to 50km/h. For $\mathrm{SNR}_o = 5, 10$ dB and $\mathrm{SNR}_t = 5$ dB, the system-rate loss and the communication burden are depicted in Fig. 7. At various speeds, simulations show that EKFUG suffers from smaller system-rate loss than LMMSEG. For example, the outage loss of EKFUG is merely 0.15% greater than that of EKFG, while LMMSEG has the highest loss of 1.4%, for a moderate speed 30 km/h, $\mathrm{SNR}_o = 10$ dB and $\mathrm{SNR}_t = 5$ dB. On the other hand, EKFUG enjoys the lowest communication burden, being only 19.6% of EKFG and 12.7% of LMMSEG. These results confirm that EKFUG is a low-complexity competitive DSA algorithm with the lowest communication burden, whereas EKFG offers the highest data rate at comparably low complexity and a moderate communication burden.

V. Conclusion

To enable practical DSA in CR networks, this paper presents Kalman filter solutions for tracking channel state information in doubly selective fading channels. Coupling with the channel tracking procedure, adaptive DSA algorithms are developed using EKF-based games, which lead to improved network spectrum utilization efficiency. The DSA decisions on the TPSD are incorporated into the state vector of an EKF, and the state-space equation is derived from the nonlinear water-filling power allocation function. The proposed EKF games, including EKFG and EKFUG, are able to jointly track the channel dynamics and update the TPSD decisions based on predicted channel information. Such a joint adaptive processing leads to improved convergence rate for iterative games, which in turn reduces the communication and computational overheads incurred by game-based DSA. The EFKUG algorithm has low communication burden and is particularly suited for applications with low-rate control channels, while the EKFG algorithm offers high spectrum utilization efficiency and is thus recommended for high-data-rate applications tolerating moderate communication overhead.

Appendix A
Derivations of the EKF Algorithm (13a)–(13c), (13e)

Using (14) into (13a), we can compute (13a). To solve (13c), (13e), we define a sign vector $\boldsymbol{\theta}_n^{(m)}$ as $[\boldsymbol{\theta}_n^{(m)}]_k = 1$, if $[\hat{\mathbf{s}}_{n|n-1}^{(m)}]_k > 0$; otherwise, $[\boldsymbol{\theta}_n^{(m)}]_k = 0$.

Define $\eta(\cdot) := \begin{pmatrix} \mathrm{Re}\{\cdot\} & -\mathrm{Im}\{\cdot\} \\ \mathrm{Im}\{\cdot\} & \mathrm{Re}\{\cdot\} \end{pmatrix}$. From (13c), we can obtain:

$$\boldsymbol{\Phi}_{E,n}^{(m)} = \begin{pmatrix} \mathbf{0}_{N_f \times N_f} & \frac{\partial \chi}{\partial \mathbf{x}_{F,n-1}^{(m)}} \\ \mathbf{0}_{2N_f N_u \times 2N_f N_u} & \frac{\partial \mathbf{x}_{F,n}^{(m)}}{\partial \mathbf{x}_{F,n-1}^{(m)}} \end{pmatrix} \tag{17a}$$

$$\boldsymbol{\Lambda}_{E,n}^{(m)} = \begin{pmatrix} \frac{\partial \chi}{\partial \mathbf{u}_{E,n}^{(m)}} \\ \frac{\partial \mathbf{x}_{F,n}^{(m)}}{\partial \mathbf{u}_{E,n}^{(m)}} \end{pmatrix}, \boldsymbol{\Psi}_{E,n}^{(m)} = \begin{pmatrix} \mathbf{0}_{N_f \times N_f} & \frac{\partial \psi}{\partial \mathbf{x}_{F,n}^{(m)}} \end{pmatrix}, \tag{17b}$$

where $(\partial \mathbf{x}_{F,n}^{(m)})/(\partial \mathbf{x}_{F,n-1}^{(m)})=\mathrm{diag}\{\eta(\mathbf{\Phi}_F^{(1m)}),\ldots \eta(\mathbf{\Phi}_F^{(N_u m)})\}$, $(\partial \mathbf{x}_{F,n}^{(m)})/(\partial \mathbf{u}_{E,n}^{(m)})=\mathrm{diag}\{\eta(\mathbf{\Lambda}_F^{(1m)}),\ldots \eta(\mathbf{\Lambda}_F^{(N_u m)})\}$, and $(\partial \psi)/(\partial \mathbf{x}_{F,n}^{(m)}) = \mathrm{diag}\{\eta(\mathbf{\Psi}_F^{(1)}),\ldots \eta(\mathbf{\Psi}_F^{(N_u)})\}$ are derived from (10) and (11).

To completely determine (17a)–(17b), the expressions for $(\partial\chi)/(\partial \mathbf{x}_{F,n-1}^{(m)})$ and $(\partial\chi)/(\partial \mathbf{u}_{E,n}^{(m)})$ are needed, which we derive as follows:

$$\begin{aligned}\frac{\partial \chi}{\partial \mathbf{x}_{F,n-1}^{(m)}} &= \left[\frac{\partial \mathbf{s}_n^{(m)}}{\partial\left|\mathcal{H}_n^{(1m)}\right|^2},\ldots,\frac{\partial \mathbf{s}_n^{(m)}}{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}\right] \\ &\quad\times \mathrm{diag}\left\{\frac{\partial\left|\mathcal{H}_n^{(1m)}\right|^2}{\partial \mathrm{Re}\{\mathbf{x}_{F,n-1}^{(1m)}\}},\right. \\ &\quad \frac{\partial\left|\mathcal{H}_n^{(1m)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{x}_{F,n-1}^{(1m)}\right\}},\ldots,\frac{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}{\partial \mathrm{Re}\left\{\mathbf{x}_{F,n-1}^{(N_u m)}\right\}}, \\ &\quad \left.\frac{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{x}_{F,n-1}^{(N_u m)}\right\}}\right\}\end{aligned} \tag{18a}$$

$$\begin{aligned}\frac{\partial \chi}{\partial \mathbf{u}_{E,n}^{(m)}} &= \left[\frac{\partial \mathbf{s}_n^{(m)}}{\partial\left|\mathcal{H}_n^{(1m)}\right|^2},\ldots,\frac{\partial \mathbf{s}_n^{(m)}}{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}\right] \\ &\quad\cdot \mathrm{diag}\left\{\frac{\partial\left|\mathcal{H}_n^{(1m)}\right|^2}{\partial \mathrm{Re}\{\mathbf{u}_{T,n}^{(1m)}\}},\right. \\ &\quad \frac{\partial\left|\mathcal{H}_n^{(1m)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{u}_{T,n}^{(1m)}\right\}},\ldots,\frac{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}{\partial \mathrm{Re}\left\{\mathbf{u}_{T,n}^{(N_u m)}\right\}}, \\ &\quad \left.\frac{\partial\left|\mathcal{H}_n^{(N_u m)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{u}_{T,n}^{(N_u m)}\right\}}\right\}\end{aligned} \tag{18b}$$

By defining $\tilde{\mathbf{B}} = \mathbf{B}\odot\mathbf{\Theta}_n^{(m)}$ and $\tilde{B}_{\mathrm{tot}} = \tilde{\mathbf{B}}^T\mathbf{1}_{N_f\times 1}$, the TPSD can be solved from (14) as

$$\begin{aligned}\mathbf{s}_n^{(m)} &= \left(\tilde{P}^{(m)}\tilde{B}_{\mathrm{tot}}^{-1}\mathbf{1}_{N_f\times 1} - (\mathbf{I}_{N_f} - \tilde{B}_{\mathrm{tot}}^{-1}\mathbf{1}_{N_f\times 1}\tilde{\mathbf{B}}^T)\right. \\ &\quad\left.\times\left((\sum_{j\neq m}|\boldsymbol{\mathcal{H}}_n^{(jm)}|^2\odot\mathbf{s}_n^{(j)} + \mathbf{n}^{(m)})\oslash|\boldsymbol{\mathcal{H}}_n^{(mm)}|^2\right)\right)\odot\mathbf{\Theta}_n^{(m)}\end{aligned} \tag{19}$$

Taking the first-order derivative on (19), it follows that (see (20) at the bottom of the page). The term $(\partial|\boldsymbol{\mathcal{H}}_n^{(jm)}|^2)/(\partial\mathrm{Re}\{\mathbf{x}_{F,n-1}^{(jm)}\})$ in (18a) can be obtained as

$$\begin{aligned}&\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Re}\left\{\mathbf{x}_{F,n-1}^{(jm)}\right\}} \\ &\quad= \left[\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Re}\left\{\boldsymbol{\mathcal{H}}_{n-1}^{(j1m)}\right\}},\ldots,\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Re}\left\{\boldsymbol{\mathcal{H}}_{n-p}^{(jm)}\right\}}\right]\Bigg|_{\mathbf{U}_{E,n}^{(m)}=\mathbf{0}}\end{aligned} \tag{21a}$$

$$\begin{aligned}&\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{x}_{F,n-1}^{(jm)}\right\}} \\ &\quad= \left[\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Im}\left\{\boldsymbol{\mathcal{H}}_{n-1}^{(jm)}\right\}},\ldots,\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\partial \mathrm{Im}\left\{\boldsymbol{\mathcal{H}}_{n-p}^{(jm)}\right\}}\right]\Bigg|_{\mathbf{U}_{E,n}^{(m)}=\mathbf{0}}\end{aligned} \tag{21b}$$

where $\boldsymbol{\mathcal{Z}}_p^{(jm)}=[\boldsymbol{\mathcal{A}}_1^{(jm)},\ldots,\boldsymbol{\mathcal{A}}_p^{(jm)}]$, $(\partial|\boldsymbol{\mathcal{H}}_n^{(jm)}|^2)/(\mathrm{Re}\{\boldsymbol{\mathcal{H}}_{n-i}^{(jm)}\})|_{\mathbf{u}_{E,n}^{(m)}=\mathbf{0}}=-2\mathrm{Re}\{\mathrm{diag}\{\hat{\boldsymbol{\mathcal{H}}}_{n|n-1}^{(jm)}\}(\boldsymbol{\mathcal{Z}}_p^{(jm)})^*\}$, and $(\partial|\boldsymbol{\mathcal{H}}_n^{(jm)}|^2)/(\partial\mathrm{Im}\{\boldsymbol{\mathcal{H}}_{n-i}^{(jm)}\}|_{\mathbf{u}_{E,n}^{(m)}=\mathbf{0}} = -2\mathrm{Im}\{\mathrm{diag}\{\hat{\boldsymbol{\mathcal{H}}}_{n|n-1}^{(jm)}\}(\boldsymbol{\mathcal{Z}}_p^{(jm)})^*\}$.

Equations (18a), (20) and (21a)–(21b) solve $(\partial\chi)/(\partial \mathbf{x}_{F,n-1}^{(m)})$. Meanwhile, for $(\partial\chi)/(\partial \mathbf{u}_{E,n}^{(m)})$, we have

$$\left.\frac{\partial\left|\boldsymbol{\mathcal{H}}_n^{(jm)}\right|^2}{\mathrm{Re}\left\{\mathbf{u}_{T,n}^{(jm)}\right\}}\right|_{\mathbf{u}_{E,n}^{(m)}=\mathbf{0}} = 2\mathrm{Re}\left\{\mathrm{diag}\left\{\hat{\boldsymbol{\mathcal{H}}}_{n|n-1}^{(jm)}\right\}\right\} \tag{22a}$$

$$\begin{aligned}&\frac{\partial \mathbf{s}_n^{(m)}}{\partial|\boldsymbol{\mathcal{H}}_n^{(jm)}|^2} \\ &\quad= \begin{cases}\mathrm{diag}\left\{\boldsymbol{\theta}_n^{(m)}\right\}(\mathbf{I}_{N_f\times N_f} - \tilde{B}_{\mathrm{tot}}^{-1}\mathbf{1}_{N_f\times 1}\tilde{\mathbf{B}}^T)\cdot diag\left\{(\sum_{i\neq m}|\boldsymbol{\mathcal{H}}_n^{(im)}|^2\odot\mathbf{s}_n^{(i)} + \mathbf{n}^{(m)})\oslash|\boldsymbol{\mathcal{H}}_n^{(mm)}|^4\right\}, & j=m; \\ -\mathrm{diag}\left\{\boldsymbol{\theta}_n^{(m)}\right\}\left(\mathbf{I}_{N_f\times N_f} - \tilde{B}_{\mathrm{tot}}^{-1}\mathbf{1}_{N_f\times 1}\tilde{\mathbf{B}}^T\right)\cdot\mathrm{diag}\{\mathbf{s}_n^{(j)}\oslash|\boldsymbol{\mathcal{H}}_n^{(mm)}|^2\}, & j\neq m\end{cases}\end{aligned} \tag{20}$$

$$\left.\frac{\partial\left|\mathcal{H}_n^{(jm)}\right|^2}{\partial \mathrm{Im}\left\{\mathbf{u}_{T,n}^{(jm)}\right\}}\right|_{\mathbf{u}_{E,n}^{(m)}=\mathbf{0}} = 2\mathrm{Im}\left\{\mathrm{diag}\left\{\hat{\mathcal{H}}_{n|n-1}^{(jm)}\right\}\right\}. \tag{22b}$$

Equations (22a)–(22b) and (18) solve $(\partial\chi)/(\partial\mathbf{u}_{E,n}^{(m)})$.

Putting all the above equations together, the *EKF algorithm* in (13a)–(13i) can be performed.

APPENDIX B
DECOMPOSITION OF EXTENDED KALMAN FILTER

To complete the decomposition, there are two steps. The first step is to decompose the EKF into a small-scale EKF and a standard KF based on $\mathbf{x}_{F,n}^{(m)}$. In the EKF Algorithm, we can rewrite $\hat{\mathbf{\Sigma}}_{E,n-1|n-1}^{(m)}$ as $\hat{\mathbf{\Sigma}}_{E,n-1|n-1}^{(m)} = \begin{pmatrix} \hat{\mathbf{\Sigma}}_{S,n-1\,|\,n-1}^{(m)} & \hat{\mathbf{\Sigma}}_{u,n-1\,|\,n-1}^{(m)} \\ \hat{\mathbf{\Sigma}}_{l,n-1\,|\,n-1}^{(m)} & \hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)} \end{pmatrix}$, where $\hat{\mathbf{\Sigma}}_{S,n-1\,|\,n-1}^{(m)}$ and $\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}$ are the updated covariance matrices of $\mathbf{s}_{n-1}^{(m)}$ and $\mathbf{x}_{F,n-1}^{(m)}$, respectively. Similarly, we rewrite $\hat{\mathbf{\Sigma}}_{E,n|n-1}^{(m)}$ as $\hat{\mathbf{\Sigma}}_{E,n|n-1}^{(m)} = \begin{pmatrix} \hat{\mathbf{\Sigma}}_{S,n\,|\,n-1}^{(m)} & \hat{\mathbf{\Sigma}}_{u,n|n-1}^{(m)} \\ \hat{\mathbf{\Sigma}}_{l,n|n-1}^{(m)} & \hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)} \end{pmatrix}$.

Based on (17a)–(17b) in Appendix A, we can obtain the following predictions on covariance matrices: Plugging (23) (shown at the bottom of the page) into (13a)–(13i), we can obtain

$$\mathbf{\Pi}_{E,n}^{(m)} = \frac{\partial\psi}{\partial\mathbf{x}_{F,n}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)}\left(\frac{\partial\psi}{\partial\mathbf{x}_{F,n}^{(m)}}\right)^H + \mathbf{\Omega}_E^{(m)} \tag{24a}$$

$$\mathbf{K}_{E,n}^{(m)} = \begin{pmatrix} \mathbf{K}_{S,n}^{(m)} \\ \mathbf{K}_{\zeta,n}^{(m)} \end{pmatrix} \tag{24b}$$

$$\hat{\mathbf{x}}_{E,n|n}^{(m)} = \begin{pmatrix} \hat{\mathbf{s}}_{n|n-1}^{(m)} + \mathbf{K}_{S,n}^{(m)}\left(\mathbf{y}_{E,n}^{(m)} - \psi\left(\mathbf{\Psi}_{E,n}^{(m)}\hat{\mathbf{x}}_{F,n|n-1}^{(jm)}, \mathbf{w}_{E,n}^{(m)}\right)\Big|_{\mathbf{w}_{E,n}^{(m)}=\mathbf{0}}\right) \\ \mathbf{x}_{F,n}^{(m)} + \mathbf{K}_{\zeta,n}^{(m)}\left(\mathbf{y}_{E,n}^{(m)} - \psi\left(\mathbf{\Psi}_{E,n}^{(m)}\hat{\mathbf{x}}_{F,n|n-1}^{(jm)}, \mathbf{w}_{E,n}^{(m)}\right)\Big|_{\mathbf{w}_{E,n}^{(m)}=\mathbf{0}}\right) \end{pmatrix} \tag{24c}$$

$$\hat{\mathbf{\Sigma}}_{E,n|n}^{(m)} = \begin{pmatrix} \hat{\mathbf{\Sigma}}_{S,n|n}^{(m)} & \hat{\mathbf{\Sigma}}_{u,n|n-1}^{(m)} - \mathbf{K}_{S,n}^{(m)} \cdot \frac{\partial\psi}{\partial\mathbf{x}_{F,n}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)} \\ \left(\mathbf{I}_{2N_uN_f\times 2N_uN_f} - \mathbf{K}_{\zeta,n}^{(m)} \cdot \frac{\partial\psi}{\partial\mathbf{x}_{F,n}^{(m)}}\right)\hat{\mathbf{\Sigma}}_{l,n|n-1}^{(m)} & \hat{\mathbf{\Sigma}}_{\zeta,n|n}^{(m)} \end{pmatrix} \tag{24d}$$

where $\mathbf{K}_{S,n}^{(m)} = \hat{\mathbf{\Sigma}}_{u,n|n-1}^{(m)}((\partial\psi)/(\partial\mathbf{x}_{F,n}^{(m)}))^H(\mathbf{\Pi}_{E,n}^{(m)})^{-1}$, $\mathbf{K}_{\zeta,n}^{(m)} = \hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)}((\partial\psi)/(\partial\mathbf{x}_{F,n}^{(m)}))^H(\mathbf{\Pi}_{E,n}^{(m)})^{-1}$, $\hat{\mathbf{\Sigma}}_{S,n|n}^{(m)} = \hat{\mathbf{\Sigma}}_{S,n|n-1}^{(m)} - \mathbf{K}_{S,n}^{(m)}(\partial\psi)/(\partial\mathbf{x}_{F,n}^{(m)})\hat{\mathbf{\Sigma}}_{l,n|n-1}^{(m)}$ and $\hat{\mathbf{\Sigma}}_{\zeta,n|n}^{(m)} = (\mathbf{I}_{2N_uN_f\times 2N_uN_f} - \mathbf{K}_{\zeta,n}^{(m)}(\partial\psi)/(\partial\mathbf{x}_{F,n}^{(m)}))\hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)}$.

Based on (23) and (24a)–(24d), and letting $\mathbf{\Pi}_{\zeta,n}^{(m)} = \mathbf{\Pi}_{E,n}^{(m)}$, it can be verified that $\mathbf{x}_{F,n-1|n-1}^{(m)}, \mathbf{x}_{F,n|n-1}^{(m)}, \mathbf{x}_{F,n|n}^{(m)}, \mathbf{\Pi}_{\zeta,n}^{(m)}, \mathbf{K}_{\zeta,n}^{(m)}$, $\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}$, $\hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)}$ and $\hat{\mathbf{\Sigma}}_{\zeta,n|n}^{(m)}$ form a standard linear KF in term of the state vector $\mathbf{x}_{F,n}^{(m)}$ that only consists of CR_m-tracked CSIs.

Summarizing, the EKF having $\mathbf{x}_{E,n}^{(m)}$ as the state vector (cf. Section III-B) has been decomposed into two parts: a standard linear KF based on $\mathbf{x}_{F,n}^{(m)}$ for channel tracking (similar to that in Section III-A) and a nonlinear EKF that updates the TPSD allocation $\mathbf{s}_n^{(m)}$ according to (15a)–(15e) in Section III-D.

It is worth pointing out that $(\mathbf{\Pi}_{\zeta,n}^{(m)})^{-1}$ used in updating $\mathbf{s}_n^{(m)}$ has already been computed inherently within the KF for updating $\mathbf{x}_{F,n}^{(m)}$. Therefore, the EKF for TPSD does not involve any matrix inversion, and thus costs much lower computational complexity than a general EKF with a same-size state vector.

The second step is to decompose the KF for $\mathbf{x}_{F,n}^{(m)}$ into N_u FDKFs. Note that $\mathbf{x}_{F,n}^{(m)}$ is formed by N_u independent real-value state vectors $\mathbf{x}_{R,n}^{(jm)} = \tilde{\mathbf{x}}_{F,n}^{(jm)}$. According to KF theory, it can be decomposed into N_u real-value KFs, each having $\mathbf{x}_{R,n}^{(jm)}$ as its state vector, $j = 1,\ldots,N_u$. The composition KF mapping (KFM) from N_u real-valued KFs to the KF for $\mathbf{x}_{F,n}^{(m)}$ is given by the first equation at the top of the next page. Conversely, the complex-to-real KF mapping (C/R-KFM) from the complex-valued FDKF Algorithm to real-valued KFs for $\{\mathbf{x}_{R,n}^{(jm)}\}$ can be derived as shown in the second equation at the top of the next page.

In all, the EKF in the game for jointly tracking CSI and TPSD, is equivalent to a EKF for TPSD, a composition KFM, N_u C/R-KFMs, and N_u FDKFs.

$$\left[\left(\hat{\mathbf{\Sigma}}_{S,n\,|\,n-1}^{(m)}\right)^T, \left(\hat{\mathbf{\Sigma}}_{u,n|n-1}^{(m)}\right)^T, \left(\hat{\mathbf{\Sigma}}_{l,n|n-1}^{(m)}\right)^T, \left(\hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)}\right)^T\right]^T = \begin{pmatrix} \frac{\partial\chi}{\partial\mathbf{x}_{F,n-1}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}\left(\frac{\partial\chi}{\partial\mathbf{x}_{F,n-1}^{(m)}}\right)^H + \frac{\partial\chi}{\partial\mathbf{u}_{E,n}^{(m)}}\mathbf{\Upsilon}_E^{(m)}\left(\frac{\partial\chi}{\partial\mathbf{u}_{E,n}^{(m)}}\right)^H \\ \frac{\partial\chi}{\partial\mathbf{x}_{F,n-1}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{x}_{F,n-1}^{(m)}}\right)^H + \frac{\partial\chi}{\partial\mathbf{u}_{E,n}^{(m)}}\mathbf{\Upsilon}_E^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{u}_{E,n}^{(m)}}\right)^H \\ \frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{x}_{F,n-1}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}\left(\frac{\partial\chi}{\partial\mathbf{x}_{F,n-1}^{(m)}}\right)^H + \frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{u}_{E,n}^{(m)}}\mathbf{\Upsilon}_E^{(m)}\left(\frac{\partial\chi}{\partial\mathbf{u}_{E,n}^{(m)}}\right)^H \\ \frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{x}_{F,n-1}^{(m)}}\hat{\mathbf{\Sigma}}_{\zeta,n-1\,|\,n-1}^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{x}_{F,n-1}^{(m)}}\right)^H + \frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{u}_{E,n}^{(m)}}\mathbf{\Upsilon}_E^{(m)}\left(\frac{\partial\mathbf{x}_{F,n}^{(m)}}{\partial\mathbf{u}_{E,n}^{(m)}}\right)^H \end{pmatrix} \tag{23}$$

$$\begin{cases} \text{Predict}: & \zeta_{x,n|n-1} = \text{vec}\left(\hat{\mathbf{x}}_{R,n|n-1}^{(1m)}, \ldots, \hat{\mathbf{x}}_{R,n|n-1}^{(N_u m)}\right), \hat{\mathbf{\Sigma}}_{\zeta,n|n-1}^{(m)} = \text{diag}\left\{\hat{\mathbf{\Sigma}}_{R,n|n-1}^{(1m)}, \ldots, \hat{\mathbf{\Sigma}}_{R,n|n-1}^{(N_u m)}\right\} \\ \text{Update}: & \mathbf{\Pi}_{\zeta,n}^{(m)} = \text{diag}\left\{\left(\mathbf{\Pi}_{R,n}^{(1m)}\right)^{-1}, \ldots, \left(\mathbf{\Pi}_{R,n}^{(N_u m)}\right)^{-1}\right\}, \mathbf{K}_{\zeta,n}^{(m)} = \text{diag}\left\{\mathbf{K}_{R,n}^{(1m)}, \ldots, \mathbf{K}_{R,n}^{(N_u m)}\right\} \\ & \zeta_{x,n|n} = \text{vec}\left(\hat{\mathbf{X}}_{R,n|n}^{(1m)}, \ldots, \hat{\mathbf{x}}_{R,n|n}^{(N_u m)}\right), \hat{\mathbf{\Sigma}}_{\zeta,n|n}^{(m)} = \text{diag}\left\{\hat{\mathbf{\Sigma}}_{R,n|n}^{(1m)}, \ldots, \hat{\mathbf{\Sigma}}_{R,n|n}^{(N_u m)}\right\} \end{cases}$$

$$\begin{cases} \text{Predict}: & \hat{\mathbf{x}}_{R,n|n-1}^{(jm)} = \tilde{\hat{\mathbf{x}}}_{F,n|n-1}^{(j)}, \hat{\mathbf{\Sigma}}_{R,n|n-1}^{(jm)} = \frac{1}{2}\eta\left(\hat{\mathbf{\Sigma}}_{F,n|n-1}^{(jm)}\right) \\ \text{Update}: & \mathbf{K}_{R,n}^{(jm)} = \eta\left(\mathbf{K}_{F,n}^{(jm)}\right), \left(\mathbf{\Pi}_{R,n}^{(jm)}\right)^{-1} = 2\eta\left(\left(\mathbf{\Pi}_{F,n}^{(jm)}\right)^{-1}\right), \hat{\mathbf{x}}_{R,n|n}^{(jm)} = \tilde{\hat{\mathbf{x}}}_{F,j}^{(n|n)}, \hat{\mathbf{\Sigma}}_{R,n|n}^{(jm)} = \frac{1}{2}\eta\left(\hat{\mathbf{\Sigma}}_{F,n|n}^{(jm)}\right) \end{cases}$$

ACKNOWLEDGMENT

The authors thank the anonymous reviewers for their constructive comments, especially on helping with the FDKFG algorithm.

REFERENCES

[1] I. J. Mitola, "Cognitive radio for flexible mobile multimedia communications," in *Proc. IEEE Mobile Multimedia Conf.*, Nov. 1999, pp. 3–10.

[2] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Select. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[3] R. Etkin, A. Parekh, and D. Tse, "Spectrum sharing for unlicensed bands," in *Proc. IEEE DySPAN Conf.*, Baltimore, MD, Nov. 2005, pp. 251–258.

[4] J. Huang, R. A. Berry, and M. L. Honig, "Distributed interference compensation for wireless networks," *IEEE J. Select. Areas Commun.*, vol. 24, no. 5, pp. 1074–1084, May 2006.

[5] W. Yu, "Competition and Cooperation in Multi-User Communication Environments," Ph.D. dissertation, Stanford Univ., Stanford, CA, 2002.

[6] W. Zhong, Y. Xu, and Y. Cai, "Capacity and game-theoretic power allocation for multiuser MIMO channels with channel estimation error," *Proc. IEEE ISCIT*, pp. 716–719, Oct. 2005.

[7] I. Barhumi, G. Leus, and M. Moonen, "Time-varying FIR equalization for doubly selective channels," *IEEE Trans. Wireless Commun.*, vol. 4, no. 1, pp. 202–214, Jan. 2005.

[8] W. Chen and R. Zhang, "Estimation of time and frequency selective channels in OFDM systems: A Kalman filter structure," *Proc. IEEE GLOBECOM*, pp. 800–803, Nov. 2004.

[9] Z. Tian and G. B. Giannakis, "A wavelet approach to wideband spectrum sensing for cognitive radios," in *Proc. 1st Int. Conf. on Cognitive Radio Oriented Wireless Networks & Coms. (CROWNCOM)*, Mykonos, Greece, Jun. 2006.

[10] M. K. Tsatsanis, G. B. Giannakis, and G. Zhou, "Estimation and equalization of fading channels with random coefficients," *Signal Process.*, vol. 53, pp. 211–229, Sep. 1996.

[11] H. S. Wang and P. C. Chang, "On verifying the first order Markovian assumption for a rayleigh fading channel model," *IEEE Trans. Veh. Technol.*, vol. 45, no. 2, pp. 353–357, May 1996.

[12] G. B. Giannakis and C. Tepedelenlioglu, "Basis expansion models and diversity techniques for blind identification and equalization of time-varying channels," *Proc. IEEE*, vol. 86, pp. 1969–1986, Oct. 1998.

[13] R. A. Iltis and A. W. Fuxjaeger, "A digital DS spread-spectrum receiver with joint channel and Doppler shift estimation," *IEEE Trans. Commun.*, vol. 39, no. 8, pp. 1255–1267, Aug. 1991.

Duo Zhang (M'06) received the B.E. and Ph.D. degrees in electrical engineering from the University of Science and Technology of China (USTC), Hefei, in 2000 and 2005, respectively.

From 2000 to 2005, he was a Research Assistant with the Personal Communication Network and Spread Spectrum (PCN&SS) Lab, School of Information Science and Technology (SIST), USTC. Since December 2005, he has been a Researcher at the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton. His current research focuses on wireless communications, particularly on software-defined radios and cognitive radios.

Zhi Tian (M'98–SM'06) received the B.E. degree in electrical engineering (automation) from the University of Science and Technology of China, Hefei, in 1994, and the M. S. and Ph.D. degrees from George Mason University, Fairfax, VA, in 1998 and 2000.

From 1995 to 2000, she was a graduate Research Assistant with the Center of Excellence in Command, Control, Communications, and Intelligence (C3I) of George Mason University. Since August 2000, she has been with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, where she is an Associate Professor. Her current research focuses on signal processing for wireless communications, particularly on ultra-wideband systems, cognitive radios and distributed sensor networking.

Dr. Tian serves as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE TRANSACTIONS ON SIGNAL PROCESSING. She is the recipient of a 2003 NSF CAREER Award.