

A DISTRIBUTED SIMULATION ENGINE FOR COOPERATIVE MULTI-USER MIMO ENHANCED WIRELESS COMMUNICATIONS

Q. Cai, Z. Zhao, H. Cao, A. Wilzeck and T. Kaiser

Institute of Communications Technology
Leibniz University Hannover, Appelstr. 9a, 30167, Hannover, Germany
phone: + (49) 511 762-2847, fax: + (49) 511 762-3030, email: cai.qipeng@ikt.uni-hannover.de
web: www.ikt.uni-hannover.de

ABSTRACT

This article addresses our concept for a highly dynamic scenario-based simulation platform for diverse wireless communication circumstance, which provides users high flexibility and ease for use to simulate complex wireless communication scenarios. It is observed that different simulation activities for wireless communication systems usually conduct a very similar processing flow. Due to this similarity the process flows could be segmented and categorized to several common tasks so that these common tasks can be easily re-used to speed up the development of new setups of simulations, such as different environment scenarios, different physical layer (PHY) parameters and even different PHY protocols, etc. However, this kind of segmentation reduces the efficiency of the computation due to the overheads of exchanging data among the tasks. To complement the loss of the computation efficiency, we structured a parallel and distributed simulation (PADS) engine to take advantage of the independence among the tasks and the high parallel computation power of the distributed computing platform (DCP).

1. INTRODUCTION

Today's wireless standards – no matter of the aimed application – provide more and more degrees of freedom to the devices and to the networks in order to find an optimal utilization of one shared resource – the channel.

The Orthogonal Frequency Division Multiplexing (OFDM) scheme is at present the basis of almost every physical layer (PHY) of broadband wireless communication standards. There are two reasons: First, OFDM is seen as the only attractive solution to cope with multi-path propagation channels in high data-rate broadband communication systems. Secondly, it is a quite elegant scheme providing a set of possibilities to adapt a broadband transmission on each of its sub-carriers to a channel of frequency selective nature. These adaptations can be carried out by selecting the sub-carriers utilized for transmission, by adjusting the power, and/or choosing the modulation and coding schemes.

Its multi-user extension, the Orthogonal Frequency Division Multiple Access (OFDMA), adds additionally, the assignment of sub-carriers to a user. Here, the number of sub-carriers and the positions can be additional part of an optimization strategy.

New wireless services and especially the wireless broadband access to the Internet with its multimedia enriched content and new data hungry applications was early generating the demand for higher data-rate, which is as of today not satisfied. Regulations of the radio spectrum available to wireless services and the scarcity of spectrum available, is always

forcing the communication engineers and researchers to develop communications technology with higher spectral efficiency. The Multiple-Input Multiple Output (MIMO) systems involving multiple antennas at receiver and at transmitter, allow transfer higher data-rate without employing additional bandwidth or transmit power. MIMO systems enable the spatial domain and thereby allow additional degrees of freedom in the utilization of the channel. Due to several technology improvements for RF transceivers and digital processing devices, MIMO systems become economically feasible with low-cost integrated circuits. The MIMO techniques are today an essential part of almost all wireless communication standards aiming to provide high data-rate communication over broadband channels, e.g. IEEE 802.11n, 3GPP LTE.

Aside from the three classical domains for adaptation algorithms: time, frequency, and code, MIMO systems add the spatial domain for the adaption method of communication system. Each domain on its own provides numerous possibilities for adaption, which leads to the challenge to find efficient ways to evaluate adaptations of links, systems, and networks for the wireless systems and networks of the future.

New paradigms of wireless communications like cooperation, multi-standard communications, cognitive radio & networks and the force to enable coexistence with other standards in the same frequency band due spectrum scarcity delivers new challenges to communication research.

At all, this requires highly dynamic scenario-based simulations for wireless communication systems with the ability to compare different solutions under equal premises.

In this contribution we introduce our concept of a simulation framework to deal with these challenges, but especially with the challenge of cooperation within multi-user environments. The rest of the paper is organized as follows. The section 2 introduces our motivation and our vision of such a framework. Our concept of the simulation platform is presented in Section 3. The section 4 explains how to describe the scenario. The central thought of task generation is given in Section 5. The section 6 defines the rules of naming the files, which helps to ease exchanging data among tasks. The section 7 illustrates the procedures of scheduling tasks and dispatching tasks to the DCP.

2. VISION

Our aim is to derive a common framework and platform for all simulations for wireless communications conducted in our institute. Main research topics of our institute are currently 3GPP LTE for indoor, distributed antenna systems, WiMax with cooperation among subscriber stations, cogni-

tive radio for heterogeneous environments involving game theory and ultra-wide band communication with multiple antennas.

Each topic provides its own challenges and requirements.

3GPP LTE operates in a licensed spectrum. Considering an indoor LTE cell, a scenario could be described with an indoor base-station and multiple mobile stations interacting in a bi-directional manner with the base-station. Downlink and uplink communication can be carried out employing Time Division Duplex (TDD) or Frequency Division duplex (FDD). Indoor LTE systems, so called femto cells, are required to operate with lower power than the normal outdoor cells of the 3GPP LTE network, but they potentially share the frequencies with the outdoor cells, however, they are not part of a network planning [1]. Hence, it is interesting to be able to model some outdoor interferers as well. The downlink of LTE involves OFDMA, the uplink SC-FDMA. Both allow different frequency assignments and bandwidth allocations.

Distributed antenna systems involve multiple differently located antennas. Objectives can be to improve the coverage of a signal within a building by emitting the same signal in different locations or to form a network of distributed transmitters or receivers and structure a so-called virtual MIMO system by cooperation among them [2].

The WiMax family of standards contains multiple PHYs meant for different applications. The IEEE 802.16d and IEEE 802.16e employ OFDMA in uplink and downlink communication and MIMO techniques are also adopted especially to extend the range and to improve the signal quality. A quite typical application for WiMax is to make provision for the broadband Internet for rural areas. Assuming subscriber stations at the edge of the coverage map an interesting study item is cooperation among subscribers with the aim to form a relaying network or with the aim to form a virtual antenna system, if another high data-rate connection among the users exist.

Cognitive radio for heterogeneous networks requires the modeling of multiple communication links employing different communication standards. A cognitive radio can sense the radio spectrum and to be aware of other communications if required. Usually, a cognitive radio is seen as secondary user with the wish to communicate with another cognitive radio, but another form is that a cognitive radio is able to re-configure itself to a different standard (potentially a primary user's standard), so that it finds for its aims the most suitable communication partner network. Dynamic Spectrum Access (DSA) [3] and Detect And Avoid (DAA) schemes [4] are important study fields, involving the ability to handle multiple bands in one simulation platform. Interesting is for example, the study of interference levels caused due to erroneous detection or prediction of unoccupied resources.

An ultra-wide band (UWB) communication is regulated to operate at 3.1 GHz to 10.6 GHz as indoor short-range communication with a low emission power. The study of multiple antennas with the aim to improve the data rate or to extend the range is an interesting study item. Multi-Standard communication devices including beside UWB; for example, an IEEE 802.11a transceiver operating in the 5 GHz ISM band and a WiMax transceiver operating in the 3.5 GHz band requires a distinct handling of the interference levels among the involved standards and UWB.

In conclusion such a simulation platform should be able to handle dynamic scenarios involving

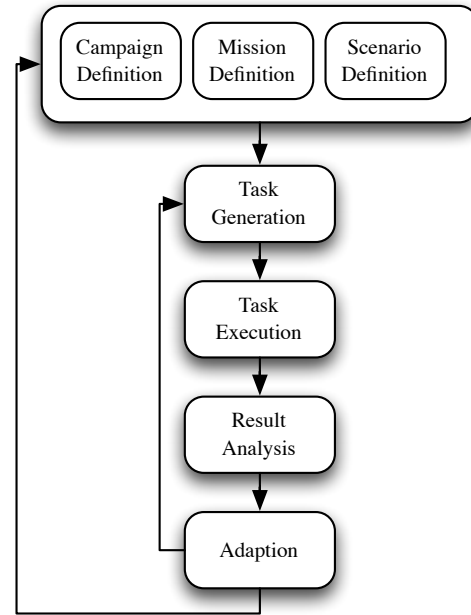


Figure 1: Simulation Cycle

- Multiple sources, no matter if user or base station, or noise,
- Multiple antennas,
- Multiple locations linked to the physical domain with help of a channel model and/or channel impulse response data-base,
- Multiple transmission intervals to study temporal behaviour of adaptations,
- Multiple frequency band intervals in base band and pass-band domain allowing to model DSA and closed loop operation in FDD mode,
- Multiple standards communications, and
- Multiple variant of involved parameters and architectures to allow the evaluation under equal conditions.

This impressive wish list requires a high degree of flexibility to a simulation framework and platform. This flexibility comes at the cost of complexity for the execution.

3. CONCEPT

A basic requirement for any new tool or framework in research is that it finds users - at least in one institute.

To handle the computational complexity induced by the rich flexibility, the computational burden is addressed by a scalable simulation engine following a parallel and distributed computing approach, which itself adds some communication overhead.

Our major goal is therefore, to hide the complexity from task scheduling and the task execution to the user. The user should be able to use the framework and the simulation engine without major modifications of his scripts and his data structures.

Fig. 1 illuminates the working steps which are conducted in a cycle manner to enable dynamic scenario-based simulations. Scenarios, missions, and campaign should be describable by the user in a natural way for a communication engineer or researcher. A campaign is thereby a set of missions,

which involve different scenarios. A mission is based on a single scenario, and it consists of the set of tasks including all variants required for comparison.

A scenario involves the definition of the scenario area, which is assumed to be provided as a grid of points. The grid points correspond to probe points in the real world (measured) or a virtual world (modeled).

In order to enable different frequency bands and different transmission time interval, we introduce a Frequency Band Identifier (FBI) and a Time Interval Identifier (TII), respectively. The user can place a transmit antenna and a receive antenna to any grid point in the scenario definition and can specify the task script, and parameter file to be called.

A task script implements a function. It is required to take one or more data files as input and to deliver one or more data files. Data are typically the signal before and after processing by the script. Parameters to the script are provided from a separate parameter file, and a script can generate as well an output parameter file for later use. To account for different programming styles and requirements, we decided not to make restrictions on the content of these files. Instead, we define only the file names to be used.

Fair comparisons involve the evaluation of different methods under equal conditions, so for example keeping the input data, the noise and the channel constant, while changing the scripts for different kinds of receivers. To reduce the effort, results of intermediate steps are stored and can be accessed later. A variant is addressed by a Variant Identifier (VI).

The execution of a campaign and its mission including complex scenarios requires the distribution of the computational burden to multiple computers operating in parallel, if possible. Therefore, after campaign, mission, and scenario definition tasks and their dependencies need to be derived. The execution of the tasks needs to be coordinated, so that the dependencies are fulfilled, intermediate results and final results are stored.

After task execution a mission and enclosed variants need to be analyzed to deploy adaptations for the next time interval, mission, or campaign.

4. SCENARIO DESCRIPTION

The key point of wireless communication system is the channel effect. Basis of a scenario is generally a data-base of Channel Impulse Responses (CIRs). This data-base can be generated by a model, a ray tracing simulation software, or data sets of a measurement campaign with a channel sounder. Currently we assume quasi-static channels for a specific time interval, so nomadic use is in focus.

In each grid point we can place a transmit antenna or a receive antenna. In addition, we can place a noise source in any grid point, no matter if it is already occupied by a transmit or receive antenna. The grid points are numbered ascendingly starting with one to provide each point with the unique identifier n_{GP} . Each grid point is linked to all others via channels which are independently described by CIRs. Since the channel can vary over time and frequency, we need to introduce a time interval and a frequency interval, in which we can assume that the channel is static. Both can be addressed with help of a TII n_{TII} and a FBI n_{FBI} . In conclusion, a channel is addressed by a grid point pair, a TII and a FBI. If transmit, noise, and incoming receive signals are generated

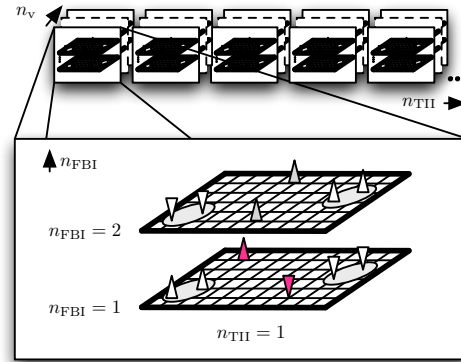


Figure 2: Example Scenario

in their corresponding grid points, it just depends on if they are active for the given TII and FBI.

The use of multiple antennas and the concept of cooperative communications (e.g. Virtual MIMO), adds the challenge to handle different spacings, different locations, spatial correlation and the involvement of much more channel impulse responses. Basically, we must be able to handle multiple transmit sources, multiple transmit points, MIMO channels with/without spatial correlation, multiple receive points, multiple receivers, and multiple noise sources in a given scenario area.

The advantage of adding the ability to handle different time intervals, frequency bands, variants, and active grid points via the corresponding identifiers, gets visible in Fig. 2. We can note that uplink and downlink communications can be handled in a natural way.

To provide fair comparisons between different parameter sets, we need the ability to freeze the situation, which means to keep channel impulse responses and to keep the generated noise samples for all variations within the TII (e.g. often identically referred as TTI). Setup variants valid for the same TII and FBI are addressed via a Variant Identifier (VI): n_v , which allows for example, to compare different Modulation and Coding Schemes (MCS) for the same channel conditions.

5. TASK GENERATION

Task generation analyzes the definition of the scenario and generates the task list for it. To reduce the task scheduling complexity and the required programming efforts by the users to adapt their code to our engine, we define for our engine only a very limited number of task classes.

They are

- Noise Source (Nx),
- Transmitter (Tx),
- Tx Superposition (TS),
- Tx Correlation (TC),
- Channel Convolution (CC),
- Rx Superposition (RS),
- Rx Correlation (RC) plus Noise Addition, and
- Receiver (Rx).

Sources are handled with highest priority, while we gave noise sources higher priority than signal sources, because

they are usually calculated fast. Noise sources are handled as special transmitters, which can be placed in already receiving or transmitting grid points. Their signals are described by a $L_{\text{TII}} \times 1$ vector $\mathbf{v}[n_{\text{TII}}, n_{\text{FBI}}, n_{\text{v}}, n_{\text{GP}}]$, where L_{TII} is the length in samples of each TII and n_{v} allows to address variants or different realizations of the signal. Corresponding to this, Tx signals are described by a $L_{\text{TII}} \times 1$ vector $\mathbf{s}[n_{\text{TII}}, n_{\text{FBI}}, n_{\text{v}}, n_{\text{GP}}]$.

Once all noise and transmit signal generation tasks are done, the superposition of signals at transmitter-side allows to add noise to the transmit signal. This Tx superposition is calculated for each grid point activated as transmitter of the current TII, FBI, and variant. For convenience and readability, we drop the currently constant index variables and place a ':' as reminder. In addition, we introduce the sets TX containing all transmitting points and Rx containing all receiving points.

The output signals of the Tx superpositions are

$$\mathbf{x}[:, n_{\text{GP}, \text{Tx}}] = \mathbf{v}^{\text{Tx}}[:, n_{\text{GP}, \text{Tx}}] + \mathbf{s}[:, n_{\text{GP}, \text{Tx}}], \quad (1)$$

where $n_{\text{GP}, \text{Tx}} \in \text{Tx}$.

Afterwards the transmit correlation in all transmit points is performed via the TC task class,

$$\mathbf{x}_{\text{c}}[:, n_{\text{GP}, \text{Tx}}] = \sum_{n \in \text{Tx}} w_{n, n_{\text{GP}, \text{Tx}}}^{\text{Tx}} \mathbf{s}[:, n], \quad (2)$$

where $w_{i,j}^{\text{Tx}}$ is a weighting factor between the j -th transmit antenna and i -th transmit antenna.

Once all Tx superpositions are calculated, the transmit signals $\mathbf{x}[n_{\text{TII}}, n_{\text{FBI}}, n_{\text{v}}, n_{\text{GP}}]$ of the current TII, FBI, and variant, are convoluted with the CIRs $h_{n_{\text{GP}, \text{Rx}}, n_{\text{GP}, \text{Tx}}}[:, :]$ valid for the links from the transmitting grid points $n_{\text{GP}, \text{Tx}}$ to the receiving grid points $n_{\text{GP}, \text{Rx}}$, generating

$$\mathbf{y}_{\text{p}}[:, n_{\text{GP}, \text{Rx}}, n_{\text{GP}, \text{Tx}}] = h_{n_{\text{GP}, \text{Rx}}, n_{\text{GP}, \text{Tx}}}[:, :] * \mathbf{x}_{\text{c}}[:, n_{\text{GP}, \text{Tx}}], \quad (3)$$

where '*' indicates the linear convolution and $n_{\text{GP}, \text{Rx}} \in \text{Rx}$.

After the convolution process, the Rx superpositions of all $n_{\text{GP}, \text{Rx}} \in \text{Rx}$ are calculated as

$$\mathbf{y}[:, n_{\text{GP}, \text{Rx}}] = \sum_{n_{\text{GP}, \text{Tx}}} \mathbf{y}_{\text{p}}[:, n_{\text{GP}, \text{Rx}}, n_{\text{GP}, \text{Tx}}]. \quad (4)$$

The follow step introduces the receive correlation effect for all $n_{\text{GP}, \text{Rx}}$ and adds the corresponding noise to yield the received signals as

$$\mathbf{r}[:, n_{\text{GP}, \text{Rx}}] = \mathbf{v}^{\text{Rx}}[:, n_{\text{GP}, \text{Rx}}] + \sum_{n \in \text{Rx}} w_{n, n_{\text{GP}, \text{Rx}}}^{\text{Rx}} \mathbf{y}[:, n_{\text{GP}, \text{Rx}}], \quad (5)$$

where $w_{i,j}^{\text{Rx}}$ is a weighting factor between the j -th receive antenna and i -th receiver antenna.

Fig. 3 sketches a 2 transmit and 2 receive point and provides insight to the calculation flow (circles are adders) and the inherent dependencies. It also points out which intermediate results can be accessed by later processing. Please note that there is no restriction that the transmit signals $\mathbf{s}(:)$ are generated by the same system (user) and that $\mathbf{r}(:)$ are employed by a single receiver.

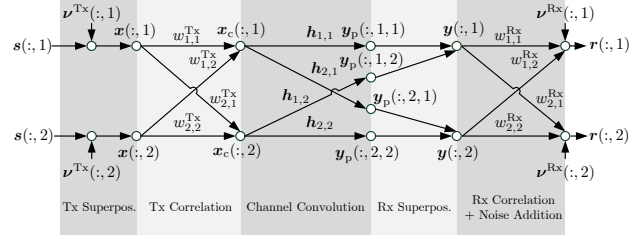


Figure 3: Calculation Flow for a 2 Transmit and 2 Receive Antenna Scenario

6. PARAMETERS & FILES

Since a lot of (intermediate) results and parameters need to be shared among the different tasks it makes sense to use a shared file server. To avoid confusion of these files a automatic naming is required.

The file naming follows the rule "`<TYPE>-<DIRECTION>-<CLASS>-<TII>-<FBI>-<POINT1>-<POINT2>-<VARIANT>.mat`", where

- TYPE: PARASET (parameter set) or DATA (data stream)
- DIRECTION: IN (input) or OUT (output)
- CLASS: TX, TS, TC, CH, RS, RC, or RX (task class)
- TII: n_{TII}
- FBI: n_{FBI}
- POINT1: $n_{\text{GP}, \text{Rx}}$
- POINT2: $n_{\text{GP}, \text{Tx}}$
- VARIANT: n_{v}

Parameter sets, and data are stored in .mat files generated by the standard functionalities of MATLAB. For each task, a valid parameter set and depending on the task class an input data file have to exist. In general, tasks can generate depending on their task class data and parameter sets, which are used as input for other tasks.

The file name "PARSET-IN-TX-1-2-0-414-3.mat" identifies for example the input (type) parameter set for the transmitter task operating in the TII $n_{\text{TII}} = 1$, for the frequency band specified by $n_{\text{FBI}} = 2$, for the grid point 414 and for the variant $n_{\text{v}} = 3$. The '0' is in case of the Tx class not evaluated.

Note that the parameter sets for the following TIIs can be generated or modified at run-time by the user simulation programs, so that for example closed-loop simulations (e.g. adaptations of the transmitters or a change of the whole PHY) can be performed. A transmitter task for example takes a parameter set as input, the executed script generates the transmit data and extracts some parameters, which are required for analysis of succeeding tasks.

7. TASK SCHEDULING & TASK DISPATCHING

Fig. 4 depicts the work flow of the system. The coordinator maintains two queues: a task queue and a worker queue. The coordinator first loads the unassigned tasks in the list to the task queue and takes care about the dependencies of the tasks. If the dependencies of a task are fulfilled and there is also an idle simulation worker in the worker queue, the coordinator assigns the task to the idle simulation worker. Beside this, it includes a watchdog timer, ensuring that a task is done within the doubled average of the assignment report time of a dedicated task class. The assignment report time is the time

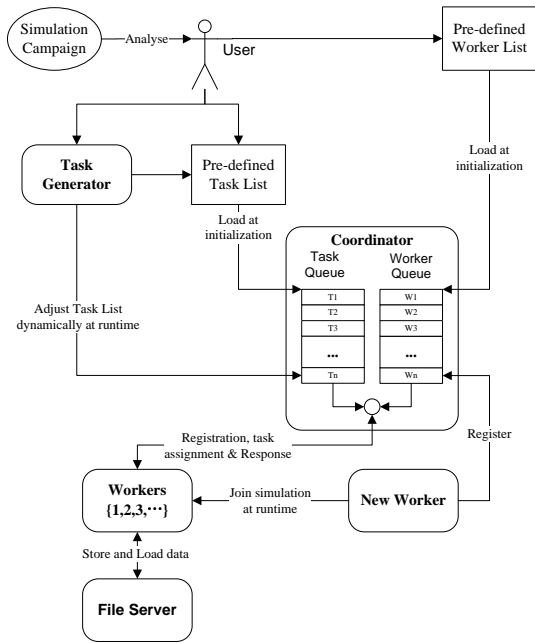


Figure 4: Working Flow of the System

difference in seconds between the moment of assignment and the moment of report.

In our system configuration, there is a fixed computer cluster dedicated to act as workers in the system. The worker list for them will be loaded to the worker queue at initialization of the engine. Users may want to add some other computers into the cluster during running of the engine in order to further speed up the simulation, if there are idle computers in the Local Area Network(LAN). Users only need to run the Matlab program for the simulation worker on the idle computers. The program will automatically register the computers to the coordinator, and the coordinator adds the new workers into the worker queue for task dispatching. The scripts to be performed are provided by the coordinator to the client, so that an update of scripts is automatically distributed.

After initialization the simulation engine starts processing the tasks in the first TII. The tasks with higher priority class will be processed earlier. NX and TX tasks have highest priority and normally, they don't need the result files from the tasks in other task classes. Therefore, NX and TX tasks in the task list will be directly assigned to the simulation workers registered in the worker queue. For the tasks of other task classes, a validation check should be executed before dispatching them to the simulation worker. If the dependencies are validated, the task will be assigned to the idle worker in the queue. If not, the coordinator checks the next task of the same task class, if its dependencies are fulfilled. All the tasks are in "UNASSIGNED" status at the beginning. When a task is assigned to a worker, its status changes to "ASSIGNED" and the associated watchdog timer starts also at the moment of assignment. The validation check and task dispatching will not be conducted for the tasks with "ASSIGNED" status in the next loop of task dispatching procedure.

When the assigned task has been done, the simulation worker sends a "FINISH" message to the coordinator. If the coordinator does not receive the "FINISH" message for the

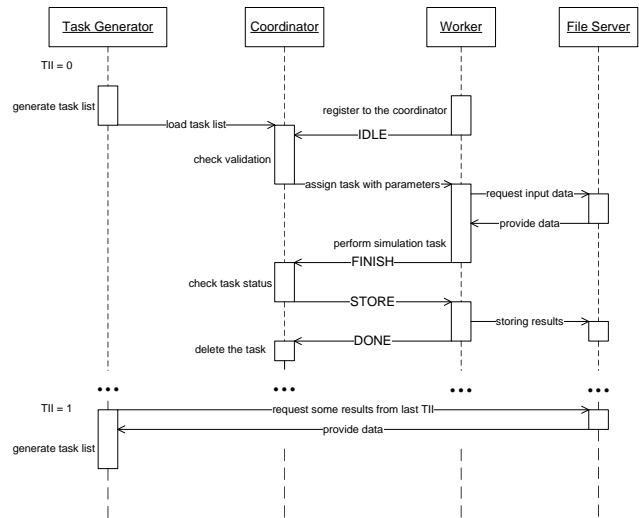


Figure 5: Interactions Flow

task before the associated watchdog timer runs out, the task's status will be changed back to "UNASSIGNED" and join in the next dispatching loop. The coordinator continues trying to dispatch the "UNASSIGNED" tasks to idle simulation workers. On the contrary, if "FINISH" message is received in time, the coordinator sends the command to the simulation worker for storing the results in the file server and delete the finished task from the task queue. If a "FINISH" message is received for a task with "UNASSIGNED" status, this can only happen when the simulation worker for the task calculates too slow to run out the watchdog timer, the coordinator will do the same process as the task has been assigned. But if the coordinator receives a "FINISH" message for the task which has been already deleted from the task queue, the coordinator will ignore this message and ask the simulation worker to stop the task. After all the tasks of the current TII have been done, the task generator can start to create the task list for the next TII. The interactions among the task generator, the coordinator, simulation workers and the file server are illustrated in Fig. 5.

REFERENCES

- [1] V. Chandrasekhar, J. Andrews, and A. Gatherer, "Femtocell networks: a survey," *IEEE Communications Magazine*, vol. 46, issue. 9, pp. 59–61, Sept. 2008.
- [2] W. Choi, J. G. Andrews "Downlink Performance and Capacity of Distributed Antenna Systems in a Multicell Environment," *IEEE TRANS. ON WIRELESS COMMUNICATIONS*, vol. 6, no. 1, pp. 69–73, Jan. 2007.
- [3] Y. Xing, R. Chandramouli, S. Mangold, and Sai Shankar N, "Dynamic Spectrum Access in Open Spectrum Wireless Networks," *IEEE J. ON SELECTED AREAS IN COMMUNICATIONS*, vol. 24, no. 3, pp. 626–636, Mar. 2006.
- [4] V. S. Somayazulu, J. R. Foerster, R. D. Roberts, "Detect and Avoid (DAA) Mechanisms for UWB Interference Mitigation," in *Proc. ICUWB 2006*, Waltham, MA, USA, September 24-27. 2006, pp. 513–518.