# Feedback from FMC users’ group

## David Lines

#### Original feedback

I've tried a quick summary below as an outline, and I'm proposing that we do an example document (with pictures) by way of clarification, but I thought useful to feedback now in time for your collation of replies.

\* We are doing a fair bit on coded excitations and we support this in our hardware/software by defining a table of waveforms.

\* We can use different waveforms on each element on each firing and we define this by an additional parameter (the index number into the waveform file) which is used in combination with the transmit delay for each element.

\* This approach allows us to support pulse inversion imaging as well as Golay-coded pair excitation, etc.

\* This also allows us to fire some clusters of elements simultaneously with other clusters using orthogonal code.

I've attached the manual that explains how we handle the above by means of a very simple extension to the industry standard Focal Law file format.

I believe that this can be incorporated into the MFMC Common File Format by defining a Waveform table. Ideally, this would be in the Root so that can be referenced within the Frame sequences, as in Paul's V1.2 changes to specification since V1.1.

Subsequent to the above, we have started looking at doing simultaneous transmissions with different (potentially orthogonal) waveforms on the same element.

These waveforms would be summed, after application of the appropriate delay for the specified waveform, to yield a new composite waveform.

Potentially we could handle this by having this composite waveform as a separate waveform in the Waveform table, but this is very inefficient and limits the flexibility of adjustment in delays.

If the delay-adjusted waveforms overlap in time, then they would only be supported by a transmitter that can handle the appropriate number of discrete voltages at any specified time, with the ultimate be an arbitrary waveform generator.

If the waveforms have delays that mean they don't overlap in time, they can be supported by current transmitter hardware.

The issue is how best to handle this extension from our current focal law specification, and it's implementation in the MFMC format.

Riliang and I have discussed this and we have some options, but I think that is best handled with pictures in a separate discussion with Paul.

Any feedback on this is welcome, but otherwise I think the next stage is for us to generate some example pictures that will help clarify the options so that we can work the best way to handle this in the MFMC specification?

MFMC for laser ultrasound systems: I'm working with Teti, and her PhD student Peter Lukacs, on the laser ultrasound FMC data that I gather you assisted in collecting.

I'm assuming that this can be defined in the MFMC Common File Format, but thought it a good time to double check?

#### Response from PW

Thanks for comments on MFMC format. I will collate feedback received and propose ways forward before next FMC users meeting, but in the meantime, can I just clarify a few points you raised:

- Coded waveforms: I assume main motivation for your interest in these is to boost SNR without averaging and because of the possibility of reducing number of transmission cycles by transmitting orthogonal codes simultaneously? The primary purpose of the MFMC file specification is to enable exchange of FMC data in a way that (a) contains the minimum information necessary to process it and (b) is as agnostic as possible to the array controller and method used to acquire it. This is different to the file format requirement of an OEM who wants to store data that is only going to be exploited by their own equipment and processing software. For this reason, I would expect OEMs to either retain their current proprietary formats, or add their own "user" datasets to the MFMC format if they wish to switch to it completely. For this reason, Martin's original suggestion for dealing with coded excitation in the MFMC specification was that the received signals should be decoded prior to storage in the MFMC\_DATA part of the MFMC file. Obviously an OEM can add extra datasets to store the undecoded data and coded excitation waveforms if they wish for their own use, but I'm not convinced that third-party software processing the FMC data needs this information. Have I missed something?

- Laser phased array: yes, MFMC format can definitely be used, although as laser PA technique is still developing the exact way MFMC file is recommended to be used might need to more closely defined in the future. For the moment though, Teti's laser PA system simply produces a frame of FMC data which can be stored as a single frame in a sequence in the MFMC file.

#### Response from DL

Thanks for the very quick feedback and you are right in your summary of some of the key application requirements.

I had thought back to the discussions with Martin & Robert (before Martin did his requirements collection phase from all interested parties), where we had wondered whether the transmission waveforms should come under the meta data for proprietary applications.

There were some applications - and Robert's interest in the Pulse Inversion Imaging for looking at kissing bonds was one such - where the excitation waveform used to acquire the data was relevant and I had thought the plan was for it to go in as an optional parameter rather than in the meta data.

Other groups in the medical research field had also expressed an interest in using the MFMC format for their raw data exchange and knowledge of the waveform was important for them too.

Since a key aim is to get more people to use the MFMC data format especially for data exchange between groups, adding additional parameters, even as an option, could be off-putting to this uptake. Perhaps the best way forward is to record it in the meta data and keep it in mind for later versions.

Thanks for confirming the position with the laser ultrasound data.

## Remi Lallement (AOS)

#### Original feedback

I talked with the people in our group who are involved in TFM-related developments. The FMC file spec looks thorough and probably covers all use cases (I say 'probably' because it is hard to think of all the possibilities). We listed a few comments and questions below, which could be discussed during the next meeting or directly with Paul if these are too detailed points.

\* 1st page, 3rd bullet point: "requirement for each frame (...) to have common parameters except the probe position" => shouldn't we also add the transmit element(s) and associated focal law in the exception list ?

\* we did not understand the paragraph which is below Figure 1. Which use case does it refer to ?

\* wouldn't it be relevant to add, at least as an optional field, some TFM grids in the file ? or the necessary parameters to reconstruct a given grid ? (i.e. frame number, position and resolution of the grid...) The interest would be that a software application loading an MFMC file could display at once the specified grid

#### Response from PW

Dear Remi,

Thanks for your comments on the MFMC format, which Tom Bertenshaw passed onto me. I will keep a list of comments received and provide an update on responses at the next FMC user meeting. In the meantime, can I clarify a few issues you raised, beginning with a brief comment on the purpose of the MFMC file spec, which I will add in some form the final document:

- The primary purpose of the MFMC file specification is to enable exchange of FMC data in a way that (a) contains the minimum information necessary to process it and (b) is as agnostic as possible to the array controller and method used to acquire it. This is different to the file format requirement of an OEM who wants to store data that is only going to be exploited by their own equipment and processing software. For this reason, the basic MFMC dataset definition is based around communicating the minimum information in the most generic form. I would expect OEMs to either retain their current proprietary formats for their internal use (or add their own "user" datasets to the MFMC format if they wish to base the internal formats on it).

With this in mind, in response to your points:

*\* 1st page, 3rd bullet point: "requirement for each frame (...) to have common parameters except the probe position" => shouldn't we also add the transmit element(s) and associated focal law in the exception list ?*

For true FMC data there is no focal law in the usual sense, since focusing (which could be adaptive and different for each frame in a sequence) is applied in post-processing at the imaging stage. The intention was that in each multi-frame sequence, FMC data in exactly the same format is acquired at each array position. I did add the option to have focal laws within the format (perhaps this is a bad idea!) due to the increasing use of, e.g. plane wave imaging where a small number of transmission cycles are used to emit plane waves at a number of different angles so it is no longer true FMC data. In current specification these would have to be the same angles for every frame in a sequence. Was there a particular scenario you were thinking about where focal laws need to change at every position?

*\* we did not understand the paragraph which is below Figure 1. Which use case does it refer to ?*

Is this the paragraph beginning "For multi-dimensional dataset ..."? If so, I will improve wording. The key point is that the MFMC sequences in a HDF5 file are designed to be indefinitely expandable - the number of frames in a sequence does not have to be defined at the outset and individual frames can be added as they are acquired (they don't have to be all added together at the end of a scan).

*\* wouldn't it be relevant to add, at least as an optional field, some TFM grids in the file ? or the necessary parameters to reconstruct a given grid ? (i.e. frame number, position and resolution of the grid...) The interest would be that a software application loading an MFMC file could display at once the specified grid*

Once again, this comes down to the purpose of the MFMC spec. Processed data (e.g. images) are intentionally excluded as the primary purpose is to enable transfer of raw FMC data between acquisition hardware and data processing software. Of course if an OEM wants to use the MFMC format as the basis of their own format they could do so by adding extra "user" datasets that contain images. However, I think the information necessary for other software to define a TFM imaging region is already present (i.e. through the ultrasonic duration of A-scans, the speed of sound, the properties of a rigid coupling wedge if used).

#### Response from RL

*For true FMC data there is no focal law in the usual sense, since focusing (which could be adaptive and different for each frame in a sequence) is applied in post-processing at the imaging stage. The intention was that in each multi-frame sequence, FMC data in exactly the same format is acquired at each array position. I did add the option to have focal laws within the format (perhaps this is a bad idea!) due to the increasing use of, e.g. plane wave imaging where a small number of transmission cycles are used to emit plane waves at a number of different angles so it is no longer true FMC data. In current specification these would have to be the same angles for every frame in a sequence. Was there a particular scenario you were thinking about where focal laws need to change at every position?*

=> RL: I obviously had PWI in mind when writing this comment, but I probably misunderstood the meaning of frame at the time. For a 64-element transducer, a frame of FMC data would consist in the 64\*64 elementary Ascans, am I right? Then in PWI a frame would be N\*64 Ascans, with N being the number of angles used. In that case my initial comment can be ignored. It is indeed relevant to have the same angles for every frame in a sequence.

*\* we did not understand the paragraph which is below Figure 1. Which use case does it refer to ?*

*Is this the paragraph beginning "For multi-dimensional dataset ..."? If so, I will improve wording. The key point is that the MFMC sequences in a HDF5 file are designed to be indefinitely expandable - the number of frames in a sequence does not have to be defined at the outset and individual frames can be added as they are acquired (they don't have to be all added together at the end of a scan).*

=> RL: Yes we were refering to this paragraph. Your explanations are very clear, thank you. It does make sense not to have to define the length of a sequence at the beginning.

\* wouldn't it be relevant to add, at least as an optional field, some TFM grids in the file ? or the necessary parameters to reconstruct a given grid ? (i.e. frame number, position and resolution of the grid...) The interest would be that a software application loading an MFMC file could display at once the specified grid

*Once again, this comes down to the purpose of the MFMC spec. Processed data (e.g. images) are intentionally excluded as the primary purpose is to enable transfer of raw FMC data between acquisition hardware and data processing software. Of course if an OEM wants to use the MFMC format as the basis of their own format they could do so by adding extra "user" datasets that contain images. However, I think the information necessary for other software to define a TFM imaging region is already present (i.e. through the ultrasonic duration of A-scans, the speed of sound, the properties of a rigid coupling wedge if used).*

=> RL: I understand your point. The use case we had in mind when making our suggestion was the following. A user "A" has recorded FMC data and processed it on his computer. He found something of interest and wants to share with a user "B" both the raw data and the reconstruted grid showing the interesting part. Storing the parameters for this grid would be convenient. But I agree this is more of a "user-friendliness" feature, and is out of the scope of purely sharing raw FMC data.