UiMOR v1.0 User Manual

Mixed-Signal Nanometer Research Lab
(www.mscad.ee.ucr.edu)

Department of Electrical Engineering,
University of California,
Riverside, CA 92521

December 6, 2011

1 Introduction to UiMOR reduction tool

UiMOR, UC Riverside Model Order Reduction Suite, is a new complexity reduction tool and was developed at Mixed-Signal Nanometer VLSI Research Lab (MSLAB) at UC Riverside.

UiMOR is a stand-along circuit complexity reduction tool. It can perform reduction for wideband frequency range with negligible loss of accuracy and is well suitable for analog/mixed-signal/memory designs. It also works well for traditional delay and noise calculations in digital circuits as well. UiMOR aims at reducing the amount of data in the netlist; reducing the memory footprint. As a result, it can speed up simulation without degrading simulation accuracy. UiMOR can take in the interconnect circuits modeled as RC/RLC circuits and produces the reduced RC/RLC circuits in SPICE format. Designers can specify the intended frequency range in which the reduced models will be accurate in its *analog* model for accurate reduction.

UiMOR is a SPICE-in, SPICE-out reduction tool and the resulting circuits are SPICE compatible. It fits seamlessly with the existing post-layout verification flow. It performs wide-band reduction for digital, analog/mixed-signal/RF circuits designs. It is extremely efficient for RC circuits with very small accuracy loss and can achieve 10-100X reduction ratio. UiMOR is very scalable and efficient for reducing interconnect circuits with million nodes.

Please contact Prof. **Sheldon Tan (stan@ee.ucr.edu)** at UC Riverside for more information and question regarding UiMOR.

2 How to run UiMOR in Linux

UiMOR currently supports the Linux computing platform. UiMOR has two reduction modes, *delay* mode and *analog* mode. *delay* model is for computing the delay of interconnects. While *analog* mode is for accurate wide-frequency modeling of parasitic interconnect modeling for given frequency range. There are two reduction options in the analog mode – *by_order*: by the reduction order and *adaptive* – by automatic reduction based on given error threshold.

The reduction is also performed for a frequency range (given or by default). The default frequency range is from $1Hz$ to 10^6Hz. This is the typically frequency range for baseband circuits.

To see the online help and usage information, simply type

```
$ ./uimor_cmd
```

Basically, the UiMOR command line has the following general form:

```
$ ./uimor_cmd -i <input_root_name> [options]
```

The "input_root_name" is a spice input file root name without the ".sp" suffix. For example, use "rc_example" instead of "rc_example.sp".

The "options" include:

- `-h` – print the usage information
- `-q <int>` – reduced order, default:
- `-m <mode>` – reduction model: 'delay' (default) or 'analog'
- `-o <string>` – output file name. by default, output name is 'input_root_name_r.sp'
- `--nsamp <int>` – the number of sampling points (in analog mode), default: 10
- `--fmax <double>` – the maximum frequency, default: 1e+06
- `--fmin <double>` – the minimum frequency, default: 1
- `--alg <alg_mode>` – the algorithm used: 'by_order' (default), 'adaptive'
- `-e <double>` – the relative error threshold used in 'adaptive' method, default: 0.1
- `-d` – perform model optimization for RLC circuits

For a given input root name, *input_root_name*, UiMOR will generate the new file named *input_root_name_r.sp* in the current directory for the resulting reduced circuit. User can specify the output file name by using `-o <string>` option.

The ports are defined by using voltage or current sources in the original circuits. The resulting reduced circuit will have the same port names and behaviors as the original circuits.

The detailed reduction options are shown in the following sections.

3 Reduction modes and examples

3.1 Delay mode reduction

The delay mode reduction has the following command:

```
$ ./uimor_cmd -i <input_root_name> [-m delay] [-q <int>]
```

If `[-q <int>]` is missed, the order is set to be 5.

For example, if we want to perform delay mode reduction on "rc_example.sp" with order 10, we have:

```
$ ./uimor -i rc_example -m delay -q 10
```

Please note that the resulting reduced model order may NOT be exactly the order assigned in the command. For most of the time, the final order of the reduced circuit will be slightly larger.

3.2 Analog mode reduction

If we want a more accurate model over a certain frequency range, we can choose analog mode reduction. There are two choices for this mode. One is "by_order", the other is "adaptive". "by_order" performs the reduction using the user assigned order. "adaptive" algorithm automatically decides the order of the reduced model based on the user assigned error threshold.

3.2.1 The "by_order" method

To perform the analog mode "by_order" reduction, use

```
$ ./uimor -i <input_root_name> -m analog [--alg by_order] [-q <int>]
[--nsamp <int>] [--fmin <double>] [--fmax <double>]
```

Please note that the default order `[-q <int>]` is 5, the number of samples `[--nsamp <int>]` is 10, `[--fmin <double>]` is 1 and `[--fmin <double>]` is 1e6.

For example, to perform analog mode "by_order" reduction on "rlc_example.sp" with order 10 and 10 samples over the frequency range [0.1, 1e7], we have:

```
$ ./uimor -i rlc_example -m analog --alg by_order -q 10 --nsamp 10
--fmin 0.1 --fmax 1e7
```

3.2.2 The "adaptive" method

For the "adaptive" reduction, use

```
$ ./uimor -i <input_root_name> -m analog --alg adaptive [-e <double>]
[--fmin <double>] [--fmax <double>]
```

Please note that the default error threshold `[-e <double>]` is 0.1, `[--fmin <double>]` is 1 and `[--fmin <double>]` is 1e6.

For example, to perform analog mode "adaptive" reduction on "rlc_example.sp" with error threshold 0.1 over the frequency range [0.1, 1e7], we have:

```
$ ./uimor -i rlc_example -m analog --alg adaptive -e 0.1 --fmin
0.1 --fmax 1e7
```

Please also note that "adaptive" method is slower than the "by_order", but it is more accurate. If the program runs too slow or generated large reduced model which is unacceptable, change the error threshold to a larger value.