This is a basic millimeter wave channel estimation simulation example with mmCEsim. The involved algorithms are ‘OMP’ and ‘Oracle LS’. There are 4 jobs in total, with SNR and pilot overhead as variables and NMSE as metric. The PFD report is auto generated via ‘simreport.cls’ and a corresponding plain text report is also available.

Contents

1 System Settings

2 Simulation Results
   2.1 NMSE v.s. SNR (Pilot: 32) .............................................................. 2
   2.2 NMSE v.s. Pilot (-10 dB) ................................................................. 3
   2.3 NMSE v.s. Pilot (0 dB) ................................................................. 4
   2.4 NMSE v.s. Pilot (10 dB) .............................................................. 5

3 Simulation Configuration
   3.1 Configuration File ................................................................. 6
   3.2 Algorithms ................................................................. 8

4 mmCEsim Information

1 System Settings

The simulation adopts the geometric channel model for millimeter wave (mmWave).

<table>
<thead>
<tr>
<th>Name</th>
<th>Antenna Number</th>
<th>Beam Number</th>
<th>Grid Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmitter</td>
<td>$8 \times 1$</td>
<td>$2 \times 1$</td>
<td>$8 \times 1$</td>
</tr>
<tr>
<td>Receiver</td>
<td>$16 \times 1$</td>
<td>$4 \times 1$</td>
<td>$16 \times 1$</td>
</tr>
</tbody>
</table>

- Channel Sparsity: 6;
- Off Grid Effect: false;
- Bandwidth: Narrowband.
2 Simulation Results

2.1 NMSE v.s. SNR (Pilot: 32)

<table>
<thead>
<tr>
<th>SNR [dB]</th>
<th>OMP</th>
<th>Oracle LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>−10</td>
<td>4.37</td>
<td>−2.09</td>
</tr>
<tr>
<td>−8</td>
<td>2.03</td>
<td>−4.57</td>
</tr>
<tr>
<td>−6</td>
<td>−0.14</td>
<td>−5.82</td>
</tr>
<tr>
<td>−4</td>
<td>−2.21</td>
<td>−8.56</td>
</tr>
<tr>
<td>−2</td>
<td>−4.45</td>
<td>−10.40</td>
</tr>
<tr>
<td>0</td>
<td>−7.81</td>
<td>−12.26</td>
</tr>
<tr>
<td>2</td>
<td>−9.78</td>
<td>−14.39</td>
</tr>
<tr>
<td>4</td>
<td>−13.08</td>
<td>−16.62</td>
</tr>
<tr>
<td>6</td>
<td>−14.69</td>
<td>−17.98</td>
</tr>
<tr>
<td>8</td>
<td>−17.75</td>
<td>−20.95</td>
</tr>
<tr>
<td>10</td>
<td>−19.39</td>
<td>−21.88</td>
</tr>
<tr>
<td>12</td>
<td>−21.63</td>
<td>−24.43</td>
</tr>
<tr>
<td>14</td>
<td>−23.87</td>
<td>−26.36</td>
</tr>
<tr>
<td>16</td>
<td>−25.62</td>
<td>−28.18</td>
</tr>
<tr>
<td>18</td>
<td>−27.74</td>
<td>−30.27</td>
</tr>
<tr>
<td>20</td>
<td>−30.37</td>
<td>−32.60</td>
</tr>
</tbody>
</table>

Simulated with 100 Monte Carlo tests.
2.2 NMSE v.s. Pilot (-10 dB)

<table>
<thead>
<tr>
<th>Pilot</th>
<th>OMP (Iter: 6)</th>
<th>OMP (Iter: 9)</th>
<th>OMP (Iter: 12)</th>
<th>Oracle LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>9.44</td>
<td>10.96</td>
<td>12.14</td>
<td>4.55</td>
</tr>
<tr>
<td>16</td>
<td>6.83</td>
<td>8.12</td>
<td>9.20</td>
<td>1.41</td>
</tr>
<tr>
<td>24</td>
<td>5.32</td>
<td>6.58</td>
<td>7.53</td>
<td>-1.07</td>
</tr>
<tr>
<td>32</td>
<td>3.78</td>
<td>5.06</td>
<td>6.06</td>
<td>-2.83</td>
</tr>
<tr>
<td>40</td>
<td>2.98</td>
<td>4.24</td>
<td>5.19</td>
<td>-3.44</td>
</tr>
<tr>
<td>48</td>
<td>2.11</td>
<td>3.36</td>
<td>4.32</td>
<td>-4.45</td>
</tr>
<tr>
<td>56</td>
<td>1.29</td>
<td>2.56</td>
<td>3.43</td>
<td>-4.73</td>
</tr>
<tr>
<td>64</td>
<td>0.90</td>
<td>2.01</td>
<td>2.81</td>
<td>-5.28</td>
</tr>
<tr>
<td>72</td>
<td>-0.06</td>
<td>1.12</td>
<td>2.06</td>
<td>-5.88</td>
</tr>
<tr>
<td>80</td>
<td>-0.28</td>
<td>0.88</td>
<td>1.80</td>
<td>-6.30</td>
</tr>
<tr>
<td>88</td>
<td>-0.82</td>
<td>0.34</td>
<td>1.19</td>
<td>-6.76</td>
</tr>
<tr>
<td>96</td>
<td>-1.28</td>
<td>-0.06</td>
<td>0.80</td>
<td>-7.50</td>
</tr>
<tr>
<td>104</td>
<td>-1.64</td>
<td>-0.27</td>
<td>0.69</td>
<td>-7.45</td>
</tr>
<tr>
<td>112</td>
<td>-2.24</td>
<td>-0.95</td>
<td>0.06</td>
<td>-8.04</td>
</tr>
<tr>
<td>120</td>
<td>-2.97</td>
<td>-1.73</td>
<td>-0.85</td>
<td>-8.52</td>
</tr>
<tr>
<td>128</td>
<td>-3.01</td>
<td>-1.63</td>
<td>-0.64</td>
<td>-8.55</td>
</tr>
</tbody>
</table>

Simulated with 200 Monte Carlo tests.
2.3 NMSE v.s. Pilot (0 dB)

<table>
<thead>
<tr>
<th>Pilot</th>
<th>OMP (Iter: 6)</th>
<th>OMP (Iter: 9)</th>
<th>OMP (Iter: 12)</th>
<th>Oracle LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>1.12</td>
<td>2.16</td>
<td>3.03</td>
<td>-5.45</td>
</tr>
<tr>
<td>16</td>
<td>-3.01</td>
<td>-1.83</td>
<td>-0.71</td>
<td>-9.34</td>
</tr>
<tr>
<td>24</td>
<td>-6.03</td>
<td>-4.54</td>
<td>-3.37</td>
<td>-11.44</td>
</tr>
<tr>
<td>32</td>
<td>-7.56</td>
<td>-5.82</td>
<td>-4.61</td>
<td>-12.38</td>
</tr>
<tr>
<td>40</td>
<td>-8.81</td>
<td>-7.04</td>
<td>-5.85</td>
<td>-13.03</td>
</tr>
<tr>
<td>48</td>
<td>-10.48</td>
<td>-8.50</td>
<td>-7.19</td>
<td>-14.37</td>
</tr>
<tr>
<td>64</td>
<td>-11.48</td>
<td>-9.29</td>
<td>-8.03</td>
<td>-15.30</td>
</tr>
<tr>
<td>72</td>
<td>-12.57</td>
<td>-10.29</td>
<td>-8.82</td>
<td>-16.24</td>
</tr>
<tr>
<td>80</td>
<td>-12.85</td>
<td>-10.49</td>
<td>-9.07</td>
<td>-16.30</td>
</tr>
<tr>
<td>88</td>
<td>-13.45</td>
<td>-11.19</td>
<td>-9.88</td>
<td>-17.02</td>
</tr>
<tr>
<td>96</td>
<td>-13.81</td>
<td>-11.50</td>
<td>-10.19</td>
<td>-17.28</td>
</tr>
<tr>
<td>104</td>
<td>-14.33</td>
<td>-11.96</td>
<td>-10.55</td>
<td>-17.90</td>
</tr>
<tr>
<td>120</td>
<td>-14.93</td>
<td>-12.50</td>
<td>-11.16</td>
<td>-18.44</td>
</tr>
</tbody>
</table>

Simulated with 200 Monte Carlo tests.
2.4 NMSE v.s. Pilot (10 dB)

<table>
<thead>
<tr>
<th>Pilot</th>
<th>OMP (Iter: 6)</th>
<th>OMP (Iter: 9)</th>
<th>OMP (Iter: 12)</th>
<th>Oracle LS</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>-5.60</td>
<td>-5.17</td>
<td>-4.48</td>
<td>-15.71</td>
</tr>
<tr>
<td>16</td>
<td>-16.00</td>
<td>-13.82</td>
<td>-12.48</td>
<td>-19.28</td>
</tr>
<tr>
<td>24</td>
<td>-17.85</td>
<td>-15.36</td>
<td>-13.74</td>
<td>-21.05</td>
</tr>
<tr>
<td>32</td>
<td>-19.06</td>
<td>-16.70</td>
<td>-15.30</td>
<td>-21.78</td>
</tr>
<tr>
<td>40</td>
<td>-19.96</td>
<td>-17.07</td>
<td>-15.51</td>
<td>-22.98</td>
</tr>
<tr>
<td>48</td>
<td>-21.25</td>
<td>-18.43</td>
<td>-16.98</td>
<td>-24.06</td>
</tr>
<tr>
<td>64</td>
<td>-23.07</td>
<td>-19.99</td>
<td>-18.37</td>
<td>-25.95</td>
</tr>
<tr>
<td>72</td>
<td>-23.60</td>
<td>-20.77</td>
<td>-19.32</td>
<td>-26.28</td>
</tr>
<tr>
<td>80</td>
<td>-23.63</td>
<td>-20.79</td>
<td>-19.31</td>
<td>-26.65</td>
</tr>
<tr>
<td>96</td>
<td>-24.79</td>
<td>-22.33</td>
<td>-21.08</td>
<td>-27.37</td>
</tr>
<tr>
<td>104</td>
<td>-25.03</td>
<td>-22.05</td>
<td>-20.58</td>
<td>-28.01</td>
</tr>
<tr>
<td>112</td>
<td>-24.87</td>
<td>-22.00</td>
<td>-20.54</td>
<td>-27.70</td>
</tr>
<tr>
<td>120</td>
<td>-25.37</td>
<td>-22.54</td>
<td>-21.07</td>
<td>-28.16</td>
</tr>
<tr>
<td>128</td>
<td>-25.49</td>
<td>-22.70</td>
<td>-21.26</td>
<td>-28.21</td>
</tr>
</tbody>
</table>

Simulated with 200 Monte Carlo tests.
3 Simulation Configuration

3.1 Configuration File

Listing 1: Example_Configuration.sim

```sim
# Example_Configuration.sim
# mmCEsim Simulation Example
# Author: Wuqiong Zhao
# Date: 2022-09-20

version: 0.1.0 # the targeted mmCEsim version
meta: # document meta data
title: mmCEsim Simulation Example
description: This is a basic millimeter wave channel estimation simulation example with mmCEsim. The involved algorithms are 'OMP' and 'Oracle LS'. There are 4 jobs in total, with SNR and pilot overhead as variables and NMSE as metric. The PFD report is auto generated via 'simreport.cls' and a corresponding plain text report is also available.

author: Wuqiong Zhao
e-mail: contact@mmcesim.org
website: https://mmcesim.org
license: MIT
date: "2022-09-18"
comments: This is an uplink channel.

physics:
frequency: narrow # assume narrow band
off_grid: false # do not consider off-grid problem

nodes:
- id: BS # this should be unique
  role: receiver
  num: 1 # this is the default value
  size: [16, 1] # UPA with size 8x4
  beam: [4, 1]
  grid: same # the same as physics size
  beamforming:
    variable: "\mathcal{W}"
    scheme: random
- id: UE # user
  role: transmitter
  num: 1 # a single-user model
  size: 8 # ULA with size 8
  beam: 2
  grid: 8
  beamforming:
    variable: "\mathcal{F}"
    scheme: random

channels:
- id: H
  from: BS
to: UE # 'from -> to' specifies the channel direction
  sparsity: 6
  gains:
    mode: normal
    mean: 0
    variance: 1

sounding:
variables:
  received: "y" # received signal vector
  noise: "noise" # received noise vector
```

channel: "H_cascaded" # the cascaded channel (actually the same as 'H' for simple MIMO)

preamble: |
COMMENT Here starts the preamble.
estimation: |
VNr::m = NEW 'DICTIONARY.R'
VNt::m = NEW 'DICTIONARY.T'

lambda_hat = INIT 'GRID.*'
Q = INIT 'MEASUREMENT' 'GRID.*'
i::u0 = LOOP 0 'PILOT'/'BEAM.T'
F_t::m = NEW F_{:,i}
W_t::m = NEW W_{:,i}

Q_{i*BEAM.*'(i+1)*BEAM.*'-1,:} = \kron(F_t^T, W_t^H) \kron(VNt^*, VNr) # the sensing matrix

END
none_zero::u1 = NEW \find(\abs(VNr^H@H_cascaded@VNt)>0.1)
# PRINT \size(\abs(\find(VNr^H@H_cascaded@VNt)),0) # make sure the number of non-zero elements
BRANCH

lambda_hat = ESTIMATE Q y none_zero
RECOVER VNr \reshape(\kron(VNt^*, VNr) # the sensing matrix)

merge: |

simulation: |
backend: c # c (default) | matlab | octave | py
metric: [NMSE] # used for compare
jobs: |
- name: "NMSE v.s. SNR (Pilot: 32)"
test_num: 100
SNR: [-10:2:20]
SNR_mode: dB # dB (default) | linear
pilot: 32
# pilot_mode: percent # num (default) | percent
algorithms: # compare different languages
- alg: OMP
  max_iter: 6
  label: OMP # used in report
  estimated_channel: H_hat_OMP # variable name for the estimated channel
- alg: Oracle_LS
  label: Oracle LS

- name: NMSE v.s. Pilot (-10 dB)
test_num: 200
SNR: -10
pilot: [8:8:128]
algorithms: # compare different languages
- alg: OMP
  max_iter: 6
  label: OMP (Iter: 6)
- alg: OMP
  max_iter: 9
  label: OMP (Iter: 9)
- alg: OMP
  max_iter: 12
  label: OMP (Iter: 12)
- alg: Oracle_LS

- name: NMSE v.s. Pilot (0 dB)
test_num: 200
SNR: 0
pilot: [8:8:128]
algorithms: # compare different languages
- alg: OMP
  max_iter: 6
  label: OMP (Iter: 6)
3.2 Algorithms

Listing 2: OMP.a1g

```matlab
# Function: OMP
# Description: Orthogonal matching pursuit compressed sensing.
# Author: Wuqiong Zhao
# Date: 2022-09-16
# Version: 0.1.0

# Input:
# - Q: Sensing matrix
# - y: Received signal
# - L: Sparsity

# Output:
# - h: The estimated sparse signal

# h: The estimated sparse signal
h::v = FUNCTION OMP Q::m y::v L::u0
# Start of OMP algorithm!
Q_H::m = NEW Q^H # the conjugate transpose of Q
r = NEW y # residual
r_last::v = NEW r * 2 # the residual in last iteration
support = INIT \length(y) dtype=u # over-length support array
term = INIT \size(Q_H, 0)$ dtype=f # float number array
j::u0 = NEW 0
# Output:
```

```matlab
FOR ** $j != \length(y)$ $j = j + 1$
    term = \abs(Q_H @ r)
    index::u0 = NEW \index_max(term)
    IF \ismember(index, support)
        h = \zeros(\size(Q, 1)) # initialize as zeros
    Q_H::m = NEW Q^H # the conjugate transpose of Q
    r = NEW y # residual
    r_last::v = NEW r * 2 # the residual in last iteration
    support = INIT \length(y) dtype=u # over-length support array
    term = INIT \size(Q_H, 0)$ dtype=f # float number array
    j::u0 = NEW 0
    a::v = INIT
```
\begin{verbatim}
BREAK # end of the LOOP
END

support_(j) = index

columns::m = NEW Q(:, support_(0:j))

a = \pinv(columns) @ y

r = y - columns @ a

IF \sum(\abs(r - r_last)) / \sum(\abs(r_last)) < 0.0001 || j >= L
  BREAK # accurate enough to end iteration
ELSE
  r_last = r
  j = j + 1
END

# prepare for the final return

h_{ support_(0:j-1)} = a

END
\end{verbatim}

Listing 3: Oracle_LS.alg

\begin{verbatim}
# Function: Oracle_LS
# Description: Oracle LS compressed sensing.
# Author: Wuqiong Zhao
# Date: 2022-09-18
# Version: 0.1.0

h::v = FUNCTION Oracle_LS Q::m y::v indices::u1
  h = \zeros(\size(Q,1))
  h_{indices} = \pinv(Q(:, indices)) @ y

END
\end{verbatim}

4 mmCesim Information

This report is auto generated by mmCesim. The application \textbf{mmCesim} is a powerful tool to simulate millimeter wave (mmWave) channel estimation (CE) for both experts and learners.

mmCesim is \textit{open source}! The software can be freely used and distributed under the MIT license.

- Official website: https://mmcesim.org
- Documentation: https://mmcesim.org/doc
- Tutorial: https://mmcesim.org/tutorial
- Web Application: https://app.mmcesim.org
- Blog: https://blog.mmcesim.org
- Publications: https://pub.mmcesim.org
- GitHub Organization: https://github.com/mmcesim
- Twitter: https://twitter.com/mmcesim