SAND2015-2860C

An Improved Deterministic Method for the Solution of Stochastic Media Transport Problems

Shawn D. Pautz
Brian C. Franke
Sandia National Laboratories
Albuquerque, NM

ANS MC2015 - Joint International Conference on Mathematics and Computation (M&C), Supercomputing in Nuclear Applications (SNA) and the Monte Carlo (MC) Method

Nashville, TN
April 19-23, 2015

SCEPTRE

Sandia National Laboratories

Exceptional service in the national interest

SAND2015-???

U.S. DEPARTMENT OF ENERGY

NNSA National Nuclear Security Administration

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000. SAND NO. 2011-XXXXP

Outline

- Background/motivation
- Previous work
- Efficiency improvements
- Results
- Conclusions

Background/motivation

We are interested in improved stochastic media computational techniques, in particular alternatives to the Levermore-Pomraning closure involving subgrid models.

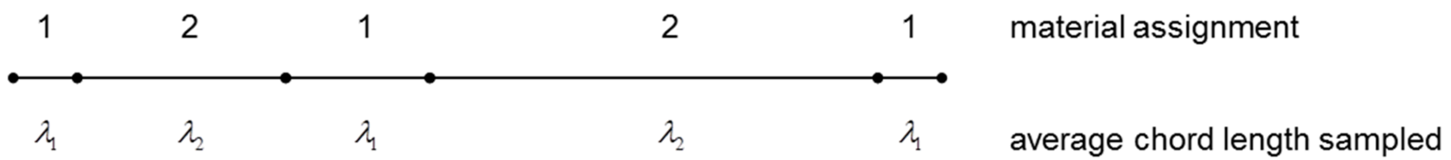
We have previously created a deterministic “sampling” technique for generating ensemble calculations.

In this work we have recognized opportunities for much greater efficiencies in our calculations.

Interface description and generation of realizations

One method for generation of realizations in 1D Markovian media directly determines interface locations

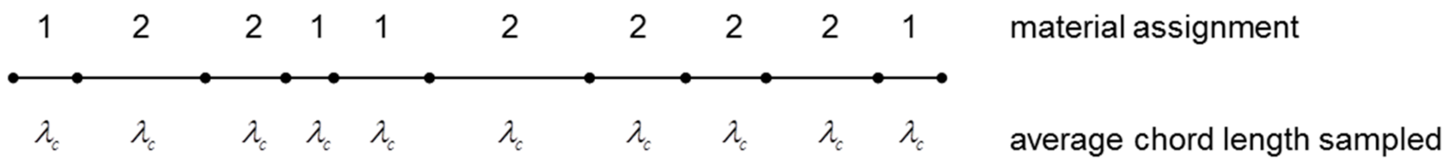
- Sample material at left boundary (e.g. material 1)
- Determine distance to first interface by sampling from $\lambda_1^{-1} e^{-\xi/\lambda_1}$, where λ_1 is the average chord length
- Determine distance to second interface by sampling from $\lambda_2^{-1} e^{-\xi/\lambda_2}$
- Repeat until far boundary reached



Pseudo-interface description and generation of realizations

A different method for generation of realizations in 1D Markovian media directly determines "pseudo-interface" locations

- Determine distance to each pseudo-interface by sampling from $\lambda_c^{-1} e^{-\xi/\lambda_c}$, where $\lambda_c = \lambda_1 \lambda_2 / (\lambda_1 + \lambda_2)$ is the combined (effective) chord length between pseudo-interfaces
- Randomly assign materials afterwards (pseudo-interfaces between identical materials disappear)



Pseudo-interface description and generation of realizations

The two descriptions/processes (interface vs. pseudo-interface) are statistically equivalent. But the pseudo-interface description has two useful properties:

- The frequency with which *P* pseudo-interfaces occur is governed by the Poisson distribution $f(P;\lambda_c) = e^{-\lambda_c}{\lambda_c}^P/P!$
- The location of pseudo-interfaces is uniformly distributed, and thus independent of the location of other pseudo-interfaces

This allows us to divide the problem into strata characterized by the number of pseudo-interfaces (with known probabilities), and also to generate realizations based on pseudo-interface location rather than region width.

Pseudo-interface approaches to the generation of realizations

- Monte Carlo sampling of the number of pseudo-interfaces, followed by Monte Carlo sampling of a realization, is equivalent to the original process
- Alternative: Stratified sampling of the number of pseudo-interfaces, followed by Monte Carlo sampling of a realization, may offer some variance reduction
- Alternative: Instead of stratified sampling, use a stratified decomposition – solve each stratified subproblem independently using the best solution technique.
    - Our approach: Within a stratum, use deterministic techniques to generate realizations instead of Monte Carlo sampling.

Deterministic generation of realizations

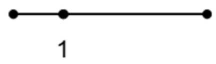
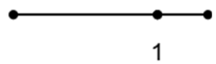
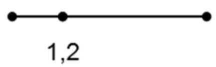
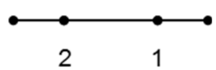
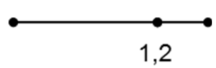
Original technique:

- Use a numerical quadrature (e.g. Gauss-Legendre) to determine the location of each pseudo-interface for a given *P*. There will be as many quadratures as pseudo-interfaces *(P*-dimensional product quadrature). The accuracy will be governed by the quadrature order(s).
- Solve the transport problem for each generated realization, and combine results according to quadrature integration rules to solve the *P*th subproblem.

$$R_P = \sum_{n_1=1}^{N} w_{n_1} \sum_{n_2=1}^{N} w_{n_2} \cdots \sum_{n_P=1}^{N} w_{n_P} \sum_{m_0=0}^{1} p(m_0) \cdots \sum_{m_P=0}^{1} p(m_P) R_{n_1 n_2 \cdots n_I m_0 \cdots m_P}$$

- Combine the results over all subproblems according to Poisson weighting: $R = \sum_{P=0}^{P_{max}} R_P f(P; \lambda_c)$

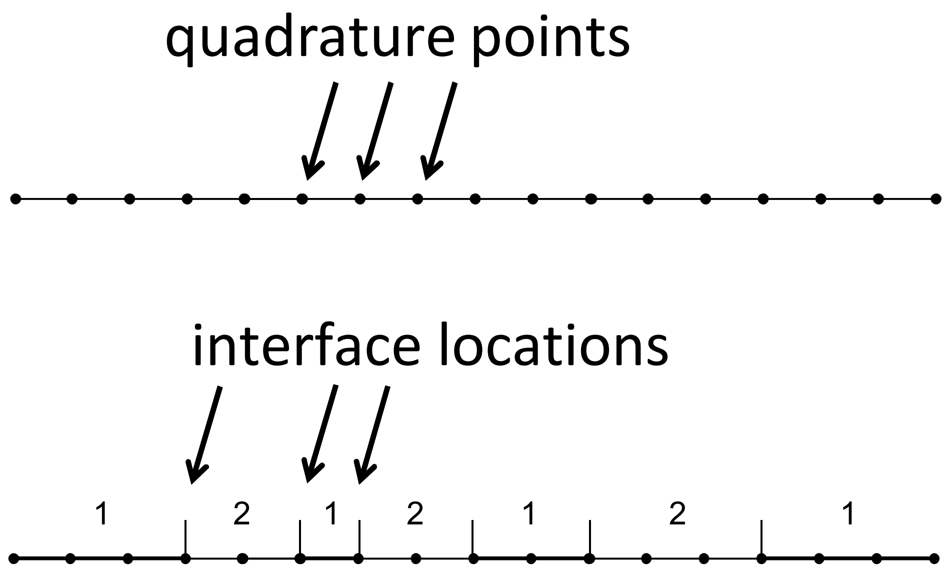
Example problem

Pseudointerfaces		Pseudointerface distribution		Material distribution		Problem
Number	Probability	Configuration	Probability	Configuration	Probability	
0	0.903924		1	1	0.9	1
				2	0.1	2
1	0.091305		0.5	1,1	0.81	1
				1,2	0.09	3
				2,1	0.09	4
				2,2	0.01	2
			0.5	1,1	0.81	1
				1,2	0.09	5
				2,1	0.09	6
				2,2	0.01	2
2	0.004611		0.25	1,1	0.81	1
				1,2	0.09	3
				2,1	0.09	4
				2,2	0.01	2
			0.25	1,1,1	0.729	1
				1,1,2	0.081	5
				1,2,1	0.081	7
				1,2,2	0.009	3
				2,1,1	0.081	4
				2,1,2	0.009	8
				2,2,1	0.009	6
				2,2,2	0.001	2
			0.25	1,1,1	0.0729	1
				1,1,2	0.081	5
				1,2,1	0.081	7
				1,2,2	0.009	3
				2,1,1	0.081	4
				2,1,2	0.009	8
				2,2,1	0.009	6
				2,2,2	0.001	2
			0.25	1,1	0.81	1
				1,2	0.09	5
				2,1	0.09	6
				2,2	0.01	2

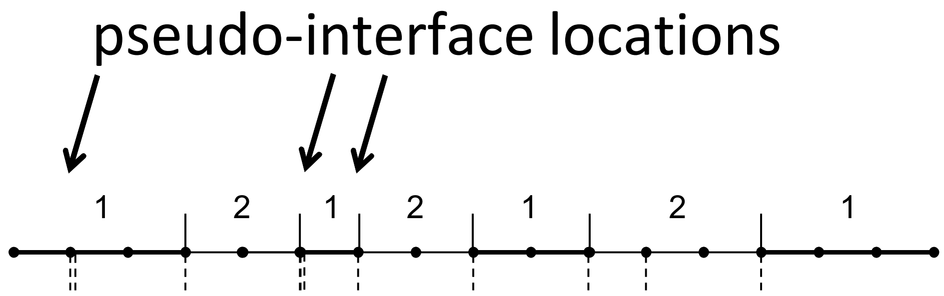
New technique

- We use a bisection rule to give equally-spaced quadrature points with equal weights.
- We generate all desired locations of interfaces to create unique transport problems
- We use counting theory to determine the frequency with which a given transport problem occurs
- Combine the results over all transport problems.

Constructing realizations



Counting realizations



N_1, N_2: number of quadrature points in material 1,2 [5,4]

n_1, n_2: number of quadrature points in material 1,2 with at least one pseudo-interface [1,1]

N_{12}, N_{21}: number of interfaces with material 1,2 on the left [3,3]

n: Total number of quadrature points with at least one pseudo-interface [8]

p_1, p_2: Probabilities of material 1,2

Counting and weighting

- Frequency with which a given transport problem occurs:

$$w_T = (wn)^p p_1^{n_1+N_{21}} p_2^{n_2+N_{12}} \frac{N_1!}{n_1!\,(N_1-n_1)!} \frac{N_2!}{n_2!\,(N_2-n_2)!} \sum_{j=0}^{n} (-1)^j \left(1-\frac{j}{n}\right)^p \frac{n!}{j!\,(n-j)!}$$

$$p_T = p_{m_l} \sum_{p=I}^{P_{max}} p_{Poisson} \sum_{n_1=0}^{\min(N_1,p-I)} \sum_{n_2=0}^{\min(N_2,p-I-n_1)} w_T$$

- Combine the results over all transport problems: $R = \sum_{T=0}^{T_{max}} R_T p_T$

Algorithmic complexity

Expense of approach for *P* pseudo-interfaces and *N*-point quadratures:

- Original algorithm: $O(2^{N+1}N^P), P \geq N$
  $O(2^{P+1}N^P), P < N$
- New algorithm : $\leq O(2^{N+1})$
- We may reduce this number further by selectively filtering out low-weight problems (problem-dependent).

Algorithmic complexity

- Number of transport calculations required for new algorithm:

N	P_{max}					
	1	2	3	7	11	15
3	8	14	16	16	16	16
7	16	58	128	256	256	256
11	24	134	464	3632	4096	4096
15	32	242	1152	32768	64384	65536

- Number of transport calculations required for new algorithm with filtering ($P_{max} = N$, benchmark problems 4-6, x=10):

N	Cumulative weight				
	0.9	0.95	0.99	0.995	0.999
3	5	7	11	14	16
7	22	42	115	135	200
11	106	250	809	1200	2092
15	333	1010	5063	7764	15871
19	883	2412	19146	34873	99275
23	2152	5884	65126	115662	477477

Results

Relative error, case 7, x=1

$\sigma_1 = \frac{2}{101}, \sigma_2 = \frac{200}{101}, c_1 = 0, c_2 = 1, \lambda_1 = \lambda_2 = 5.05$ (P_{avg}=0.396)

Reflection:

N	P_{max}					
	1	2	3	7	11	15
3	0.01	0.01	0.01	0.01	0.01	0.01
7	< 0.01	< 0.01	0.01	0.01	0.01	0.01
11	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01
15	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01	< 0.01

Transmission:

N	P_{max}					
	1	2	3	7	11	15
3	-0.002	-0.003	-0.003	-0.003	-0.003	-0.003
7	-0.001	-0.002	-0.002	-0.002	-0.002	-0.002
11	< 0.001	-0.001	-0.001	-0.001	-0.001	-0.001
15	< 0.001	-0.001	-0.001	-0.001	-0.001	-0.001

Results

Relative error, case 4, x=1

$$\sigma_1 = \frac{10}{99}, \sigma_2 = \frac{100}{11}, c_1 = 0, c_2 = 1, \lambda_1 = 9.9, \lambda_2 = 1.1, (P_{avg}=1.01)$$

Reflection:

N	P_{max}					
	1	2	3	7	11	15
3	-0.03	0.02	0.02	0.03	0.03	0.03
7	-0.05	0.01	0.02	0.02	0.02	0.02
11	-0.06	< 0.01	0.01	0.01	0.01	0.01
15	-0.06	< 0.01	0.01	0.01	0.01	0.01

Transmission:

N	P_{max}					
	1	2	3	7	11	15
3	0.005	-0.002	-0.003	-0.004	-0.004	-0.004
7	0.007	-0.001	-0.002	-0.003	-0.003	-0.003
11	0.008	< 0.001	-0.002	-0.002	-0.002	-0.002
15	0.009	0.001	-0.001	-0.002	-0.002	-0.002

Results

Relative error, case 1, x=1

$\sigma_1 = \frac{10}{99}, \sigma_2 = \frac{100}{11}, c_1 = 0, c_2 = 1, \lambda_1 = 0.99, \lambda_2 = 0.11$, ($P_{avg}$=10.1)

Reflection:

N	P_{max}					
	1	2	3	7	11	15
3	-0.423	-0.325	-0.266	-0.182	-0.171	-0.170
7	-0.437	-0.325	-0.247	-0.090	-0.046	-0.039
11	-0.443	-0.330	-0.248	-0.075	-0.021	-0.011
15	-0.445	-0.333	-0.251	-0.071	-0.013	-0.003

Transmission:

N	P_{max}					
	1	2	3	7	11	15
3	0.128	0.098	0.081	0.055	0.052	0.051
7	0.133	0.099	0.075	0.027	0.014	0.012
11	0.134	0.100	0.075	0.023	0.006	0.003
15	0.135	0.101	0.076	0.022	0.004	0.001

Results

Relative error, case 1, x=10

$$\sigma_1 = \frac{10}{99}, \sigma_2 = \frac{100}{11}, c_1 = 0, c_2 = 1, \lambda_1 = 0.99, \lambda_2 = 0.11, (P_{avg}\text{=101})$$

Reflection:

N	P_{max}					
	1	2	3	7	11	15
3	-0.735	-0.696	-0.671	-0.632	-0.624	-0.623
7	-0.734	-0.683	-0.643	-0.544	-0.497	-0.473
11	-0.734	-0.681	-0.636	-0.516	-0.448	-0.406
15	-0.734	-0.680	-0.634	-0.504	-0.425	-0.374

Transmission:

N	P_{max}					
	1	2	3	7	11	15
3	4.7	4.4	4.2	3.8	3.8	3.8
7	4.7	4.3	4.0	3.2	2.9	2.7
11	4.7	4.3	3.9	3.0	2.5	2.3
15	4.7	4.3	3.9	3.0	2.4	2.0

Filtered results

Relative difference between filtered and unfiltered results, case 4, x=10 (P_{avg}=10.1, P_{max}=N)

Reflection:

N	Cumulative weight				
	0.9	0.95	0.99	0.995	0.999
3	-0.361	-0.180	-0.044	-0.013	0
7	-0.193	-0.110	-0.023	-0.013	-0.002
11	-0.154	-0.083	-0.018	-0.010	-0.002
15	-0.149	-0.069	-0.015	-0.008	-0.002

Transmission:

N	Cumulative weight				
	0.9	0.95	0.99	0.995	0.999
3	0.101	0.047	0.010	0.003	0
7	0.084	0.048	0.008	0.004	< 0.001
11	0.094	0.045	0.009	0.005	< 0.001
15	0.100	0.040	0.008	0.004	< 0.001

Relative expense:

N	Cumulative weight				
	0.9	0.95	0.99	0.995	0.999
3	0.313	0.438	0.688	0.875	1
7	0.086	0.164	0.449	0.527	0.781
11	0.026	0.061	0.198	0.293	0.511
15	0.005	0.015	0.077	0.118	0.242

Conclusions

- New approach is significantly more efficient than the previous algorithm
- Accuracy is inversely related to P_{avg}
- Errors decrease as both P_{max} and N increase
- In many cases the method is less expensive than Monte Carlo sampling
- Filtering can substantially reduce the expense even more with little degradation in results