

分割構築によるフロンティア法の省領域化

齋藤 寿樹

神戸大学大学院工学研究科電気電子工学専攻



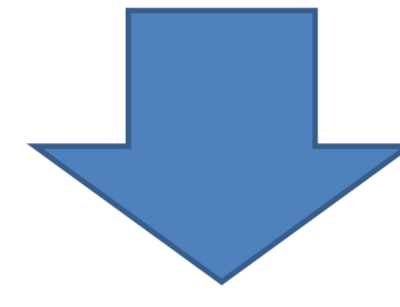
フロンティア法

- BDD/ZDD/MvDDをトップダウンに構築する手法
 - s-tパス, サイクル, 木(森)など
- フロンティア法で設計すること
 - 共有化処理
 - 枝刈り処理
 - 上記処理を効率的に行うためのデータ構造

メモリ不足による計算に限界

フロンティア法の限界

フロンティアサイズが大きい



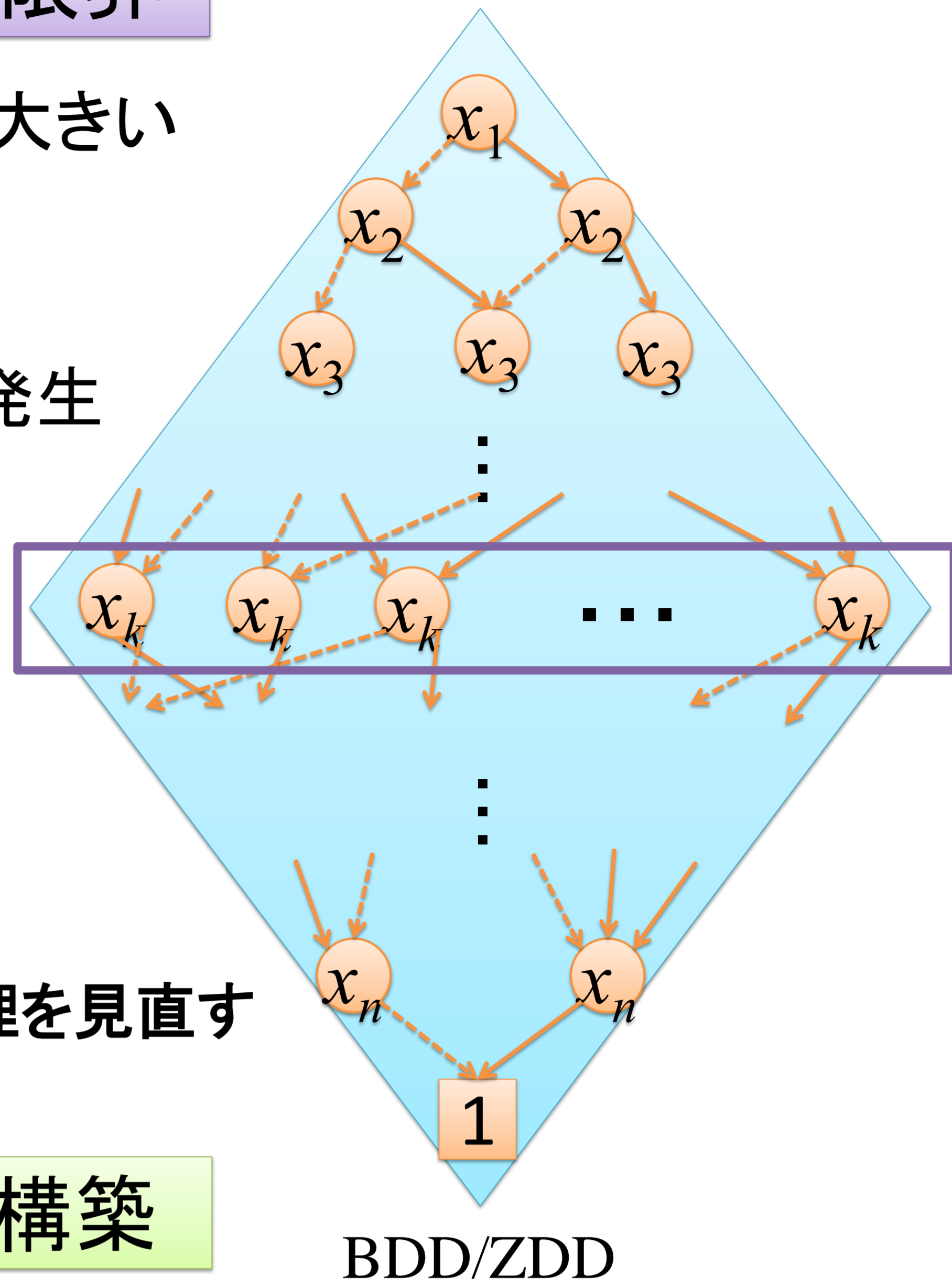
膨大な数のノードが発生

メモリ上に格納できない

解決方法

- ・変数順序を変える
- ・共有化処理, 枝刈り処理を見直す などなど

1つの方法: 分割構築



分割構築

アルゴリズム

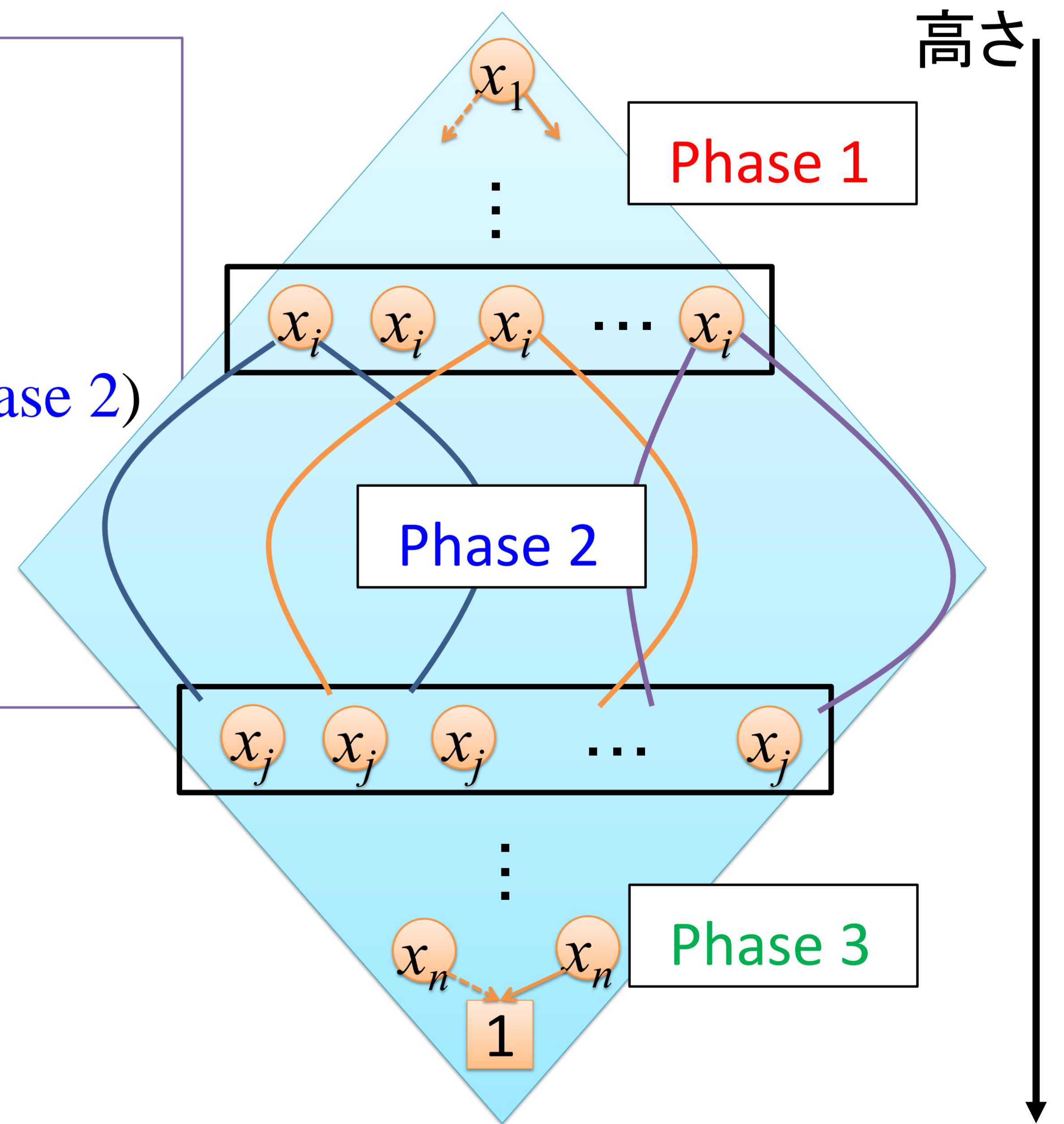
1. フロンティア法を高さ i まで実行(Phase 1)
2. 高さ i の各ノード v に対して
3. ノード v を根として, フロンティア法を高さ j まで実行(Phase 2)
4. 高さ j からフロンティア法を実行(Phase 3)

分割構築中のフロンティア法で最大幅のすべてのノードをメモリ上に蓄えられなかったら, 再度分割構築を実行する

メモリ上に格納するノード数を削減できる



メモリ不足による計算限界を回避



計算機実験

あみだくじの数え上げ[中嶋ら, 2014]

分割開始高さ2のMDDノード数

n	Bars	MDD size			#Ladders	Time [sec]
		(1st Phase)	(2nd Phase)	(3rd Phase)		
5	10	9	10	10	64	2.35
6	15	14	55	40	908	3.20
7	21	20	347	73	24698	4.22
8	28	27	2588	303	1232944	5.74
9	36	35	21743	530	112018190	10.95
10	45	44	205291	2263	18410581880	71.53
11	55	54	2150770	9640	5449192389984	1094.43
*13	78	77	982716795	40333	2752596959306389652	451889.11

Phase 3の開始高さは $n/3$ で固定

縦線本数9

i	MDD size (1st Phase)	MDD size (2nd Phase)	time
-	65471	0	2.98
1	8	32341	2.11
3	117	16489	28.24
5	713	11375	137.02
7	2706	8972	482.17
9	7523	3822	1302.75

100GB -> 22GB

使用メモリ量: 約1/6
計算時間: 約437倍

まとめと今後の課題

まとめ

- ・分割構築を利用したフロンティア法を提案
- ・メモリ使用量を削減して数え上げを行える
- ・並列化が容易

今後の課題

- ・他の問題への適用
- ・Phase 2の高速化
 - ・並列化
 - ・演算キャッシュ
- ・メモリ使用量と計算時間のトレードオフの解析
- ・厳密にBDD/ZDD/MvDDを構築するには?