

УДК 004.89

## ПАРАЛЛЕЛЬНЫЕ АЛГОРИТМЫ ОПТИМИЗАЦИИ ГРАНИЦ КАРЬЕРОВ ПО МЕТОДУ ПСЕВДОПОТОКА НА МОДЕЛИ ДАННЫХ СО СТРУКТУРОЙ ОКТОДЕЕРЕВА

## PARALLEL OPTIMIZATION ALGORITHMS OF OPEN PIT LIMITS USING PSEUDO FLOW METHOD ON BLOCK MODEL WITH OCTREE STRUCTURE

**П.В. Васильев, В.М. Михелев, Д.В. Петров**  
**P. V. Vassiliev, V. M. Mikhelev, D. V. Petrov**

*Белгородский государственный национальный исследовательский университет,  
Россия, 308015, Белгород, ул. Победы, 85*

*Belgorod State National Research University, 85 Pobeda St, Belgorod, 308015, Russia*

*e-mail: vassiliev@bsu.edu.ru, mikhelev@bsu.edu.ru, petrov@bsu.edu.ru*

*Аннотация.* В статье рассмотрен метод максимизации сетевого псевдопотока для решения задачи поиска предельных границ карьеров. Существующий метод предполагает выполнение расчетов на регулярных блочных моделях с одинаковыми размерами ячеек решетки. В новой модификации метода для ускорения расчетов предлагается использовать также блочные модели со структурой октодерева. Это существенно сокращает время выполнения расчетов на сложных и сверхбольших блочных моделях. В этой связи в интегрированную систему недропользования Geoblock введено распараллеливание вычислений в модули с циклами интерполяции и оптимизации. Тестирование и проверка эффективности разработанной схемы алгоритма выполнялись на наборах моделей стандартной библиотеки MineLib.

*Resume.* The article describes parallel modification for of known pseudo flow method with high label for design optimum open pit. It is supposed that regular block models with uniform grids are using in the existing method. In the new modification of this method also block models with octree structures could be used. That should further reduce the time of calculations on complicated and huge block models. In this connection the interpolation and optimization circuits were modified for parallel calculations in units of Geoblock integrated software. For testing and verification of the new algorithm scheme the block models are applied from well-known MineLib library datasets.

*Ключевые слова:* блочное моделирование, структура октодерева, оптимизация карьеров, алгоритм псевдопотока, параллельные вычисления

*Keywords:* block modelling, octree structure, open pit optimization, pseudo flow method, parallel computations

Анализ современных методов оптимизации границ карьеров по добыче руд [Васильев, 2015] и их сопоставление по вычислительной сложности показывает, что для поиска наиболее выгодных с экономической точки зрения оболочек карьеров наиболее эффективными являются алгоритмы теории графов и, в частности, алгоритмы, основанные на нахождении максимальных потоков в ориентированном графе блочной модели месторождения. Ряд алгоритмов орграфов и их обработка приведены, в частности, в [Ахо, 2000]. Для регулярных блочных моделей общая схема может быть представлена в виде графа, показанном на рис. 1.

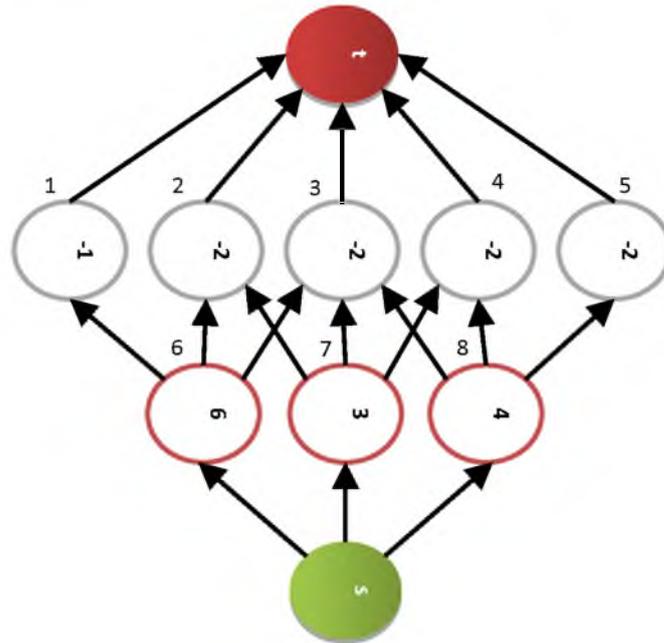


Рис. 1. Орграф сетевого потока от источника s к стоку t по регулярной решетке блоков  
Fig. 1. Digraph network flow from the source s to the sink t on a regular lattice of blocks

**Построение октарной блочной модели для оптимизации**

Схема структуры октодеревя пространства месторождения, состоящего из однокомпонентной руды (черное) и пустой породы (белое), в виде октодеревя представлена на рис. 2.

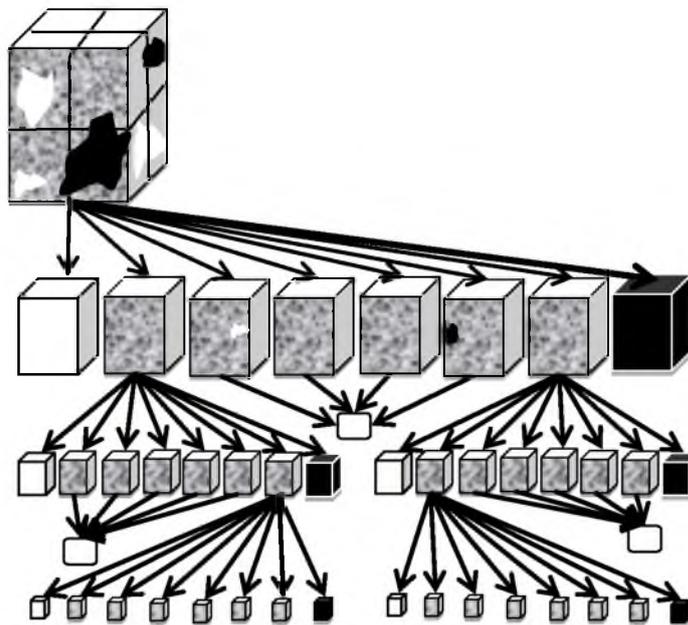


Рис. 2. Структура октоэдрического дерева модели  
Fig. 2. Octahedral structure model tree

Серые области на самом деле являются текстурой, состоящей из белых и черных тел или зерен. Алгоритм формирования пирамидально рекурсивной структуры октодеревя состоит в следующем. Цвет каждого из  $N$  блоков может принимать значение 0 (w - "белое", порода), 1 (b - "черное", рудный минерал) и 2 (g "серое", руда+порода). На нулевом уровне структуры исходный блок (суперблок) считается одним серым фрагментом со средним содержанием рудного компонента  $C_0$ . На первом уровне объем суперблока нулевого уровня разбивается на  $k$  равных частей (для октодеревя  $k=8$ ) и мы получаем  $k$  фрагментов 1-го уровня разбиения. На втором уровне число фрагментов равно  $k^2$ . Каждому фрагменту сопоставляется среднее содержание рудного минерала  $C(i,j)$ ;  $i,j=0,k-1$ . Повторяя эту процедуру рекурсивно  $t$  раз получим фрагменты  $t$ -го разбиения, совпадающие с минималь-



ной крупностью блока селективной добычи БСД со средним качеством  $C^t(i,j)$ ;  $i,j=0,k^t-1$ . Для всей структуры формула имеет вид:

$$C^{t+1}(i,j) = \frac{1}{k^2} \sum_{l=0}^{k-1} \sum_{n=0}^{k-1} C^t(ki+l, kj+n) \quad (1)$$

Нетрудно показать, что качество каждого блока равно среднему качеству составляющих его подчиненных блоков (квантов) любого нижележащего уровня. В нашем случае качество блока есть доля полезной минеральной фазы в нем. Разбиение исходной полигональной модели (состоящей из полиэдров Вороного, оболочек рудных тел) с помощью пирамидально-рекурсивного алгоритма имитирует процесс сокращения крупности блоков (участков месторождения, кусков или частиц) и позволяет вычислить спектры распределения блоков каждого уровня по крупности по качеству.

Распределение средних содержаний блоков одного уровня разбиения представляет собой спектр раскрытия белых (породы) и черных (руды) блоков соответствующего класса крупности. Подобная усеченная структура представляется в виде несбалансированного октодеревя. Все терминальные вершины этого дерева белые или черные, а нетерминальные - серые. С технологической точки зрения построение оптимальной усеченной структуры октодеревя эквивалентно тому, что при дроблении массива руды на каждой стадии фрагментируются лишь те куски (серые фрагменты) из данного класса крупности, которые состоят из двух или более фаз, монофазные же фрагменты, т.е. чистая пустая порода или чистый рудный минерал, далее не дробятся. Кроме того, это позволяет оценить число вершин различных "цветов" в октодереве и предсказать объемную долю руды, которую целесообразно измельчать дальше в каждом классе крупности для ОМГТ. Среднее число вершин различных типов определяется в зависимости от номера уровня  $t$  формулами:

$$w(t) = b(t) = k^t(k - \frac{1}{2}); g(t) = k^t; t = 1, 2, \dots, T; \quad (2)$$

где  $w(t), b(t), g(t)$  - соответственно число белых, черных и серых вершин на  $t$ -м уровне для простого сростка при  $s_{cp} = 50\%$ . Число серых фрагментов пропорционально площадной доле сростков, появившихся в процессе сокращения крупности кусков руды после операции разделения качественного материала и пустой породы. До определенного уровня дробления  $t_0$  практически все вершины дерева остаются "серыми", а затем появляются полностью раскрытые блоки фаз.

При горнопромышленном освоении сложных месторождений твердых полезных ископаемых и решении задач управления запасами минерального сырья размеры каркасных и блочных геологических моделей могут достигать больших размеров. Создание совместных геолого-маркшейдерских моделей и систем управления качеством руд требует постоянного увеличения потоков обработки данных. В результате на время подсчета запасов и определение наилучших конфигураций выемки могут уходить многие часы работы быстродействующих компьютеров. В связи с этим становится актуальной проблема выбора структуры хранения компьютерной модели месторождения и поиска наиболее эффективных методов оптимизации ведения горных работ.

Представление цифровой геологической модели в виде решетки блоков или массива прямоугольных ячеек позволяет использовать весь комплекс методов геостатистики и достаточно просто оценивать запасы. Однако присутствие в массиве больших однородных областей, характеризующихся слабой анизотропией геопоказателей, никак не уменьшает объем моделей. Вместе с тем структура октодеревя позволяет компактно хранить и с высокой скоростью обрабатывать большие массивы данных. Тем не менее, данная структура, также, как и регулярная матрица вокселей, не позволяет представить отдельно от вещественной составляющей поверхности пластов, линии складчатости и разломы.

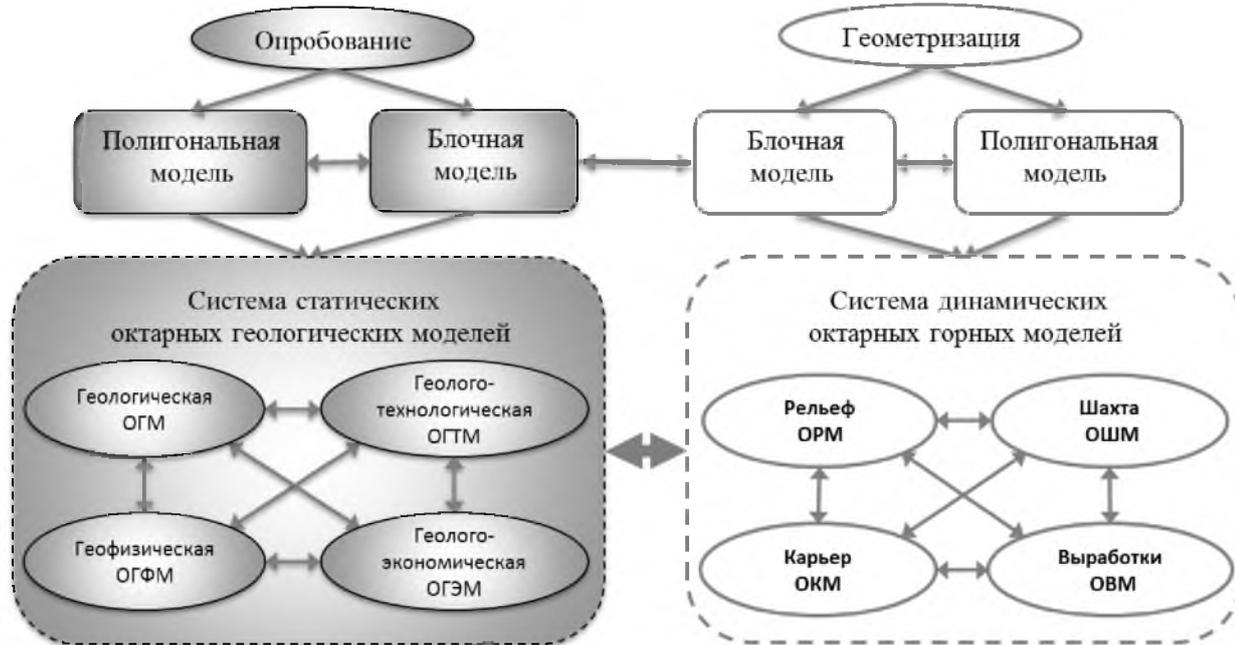


Рис. 3. Система блочных моделей со структурой октодеревя для решения задач оптимизации в недропользовании  
 Fig. 3. The system block models with the structure of octree for solving optimization problems in the subsurface

### Модификация алгоритма псевдопотока для структуры данных в виде октодеревя

Как отмечается в работе [Muir, 2008] нормализованные деревья алгоритма LG были адаптированы к более общей модели сетевого потока на основе концепции псевдопотока, по аналогии с предпотоками (preflow). Сетевой псевдопоток удовлетворяет ограничениям пропускной способности, но в нем могут нарушаться условия баланса потока с созданием дефицита или избытка в узлах. Предпоток удовлетворяет ограничениям пропускной способности, но в нем может нарушаться баланс потока лишь путем создания избытка в узлах. Алгоритм псевдопотока решает задачу нахождения максимального потока в обобщенных сетях и работает с псевдопотоками вместо масс.

Связь между алгоритмом псевдопотока и алгоритмом предпотока (переразметки, push-relabel) более очевидна, чем между алгоритмом LG и алгоритмом предпотока. Методы LG и псевдопотока имеют дело с множеством узлов (ветвей), способных аккумулировать либо избыток, либо дефицит. В подходе на основе псевдопотока, массы  $M_r$ , поддерживаемые корневым узлом  $r_0$  сильного дерева, трактуются как некий псевдопоток и выталкиваются в слабый корень  $r_s$ , а затем к фиктивному корневному узлу  $x_0$  (одновременно являющемуся как источником, так и стоком). Алгоритм выталкивания с переразметкой работает с предпотоками (preflows). Алгоритм переразметки работает скорее с самими узлами, а не с множествами узлов, и избыток в некотором узле выдавливается в находящиеся ближе к стоку узлы в соответствии со значениями дистанционных меток, переразметка обновляет значения меток.

Алгоритм псевдопотока предоставляет несколько способов обработки вершин типа «слабый-над-сильным». Лучшими из этих методов являются варианты с нижней и верхней метками [Hochbaum, 2008]. Методы нижней и верхней меток работают с концепцией дистанционной метки. Для некоторого узла дистанционная метка представляет собой неубывающую функцию и на этом уровне она является неубывающей в составе любого сгенерированного дерева. В статье [Hochbaum, 2008] доказывается, что дистанционная метка есть неубывающая функция на заданном уровне дерева и для слабого узла  $v$  она является некоторой нижней границей уровня ( $v$ ). Функция дистанционной метки аналогична дистанционным меткам, введенным Голдбергом и меткам, используемым в методах сетевого потока, таких как метод переразметки [Ахоб 2000]. Так, метод Голдберга-Тарьяна дает оценку сложности для максимального потока  $O(nm \log(n^2/m))$ . В методе же псевдопотока при первоначальной нормализации вложенного остовного дерева  $T_v$ , всем сильным узлам блоков присваивается метка 2, а всем слабым узлам присваивается метка 1. Для эффективного управления сильными ветвями создается и поддерживается приоритетная очередь с индексом. Счетчик отслеживает количество сильных корневых узлов и индексированный список, указывающий на первый сильный корневой узел, для которых вводятся метки (первоначально все имеют индекс 0).

Модифицированный алгоритм псевдопотока на структуре октодерева выполняется в следующей последовательности:

Шаг 1. Определяется тип данных структуры блочной модели.

Шаг 2. Первоначально все положительные узлы считаются сильными и являются, соответственно, корнями своих ветвей. На этой стадии определяется индекс блока, его принадлежность к уровню и ветви дерева.

Шаг 3. Если после размер блока таков, что угол наклона борта превышает допустимый, то целый текущий блок разбивается на следующую восьмерку блоков меньшего размер, пока угол наклона борта не будет лежать в допустимых пределах.

Шаг 4. Выполняется расстановка меток. Всем таким узлам присваивается метка 2 и они помещаются в очередь. Порядок произвольный, поскольку все сильные узлы имеют пометку 2, хотя фактический порядок определяет последовательность обработки ветви. Указатель на первый узел с меткой 2 заносится в индексированный список.

Шаг 4. При выборе для обработки следующего сильного дерева, верх очереди (при упорядочивании либо сверху, либо снизу) выбирается и удаляется из очереди.

Шаг 5. После того, как процессы слияния и нормализации генерируют новую сильную ветвь, то она вставляется в очередь в верхнюю позицию для этой метки.

Выполнение параллельной оптимизации было реализовано с целью получения максимальной выгоды от работы многоядерных компьютеров при работе как на настольных, планшетных, так и на мобильных устройствах. Для распараллеливания циклов в программной системе использовалась последняя версия библиотеки у параллельных вычислений Parallel Programming Librar в составе RAD Studio. Модуль System.Threading упрощает процесс распараллеливания сложных вычислительных задач и предлагает набор необходимых команд, встраиваемых в новые и уже существующие проекты, написанные на языке C++. В код также включена процедура автоматической автонастройки пула потока, оценивающая загруженность CPU и дополнительные опции для тонкой настройки многопоточных вычислений. Таким образом, в системе недропользования Geoblock была выполнена замена оператора "for" на обращение к class function TParallel->For, передаваемой в код на выполнение в качестве анонимного метода. Кроме того, для исключения вероятных конфликтов в исполнении многопоточных вычислений, стандартное приращение i++ было заменено командой TInterlocked->Increment. Общие циклы процедур интерполяции показателей и оптимизации для больших массивов исходных точек данных были преобразованы согласно следующим конструкциям:

```
TParallel->For(1; N; void (int I)
{
    if (IsInterpolated (I))
        TInterlocked->Increment (Count);
});
и
TParallel->For(1; N; void (int I)
{
    if (IsOptimized (I))
        TInterlocked->Increment (Count);
});
```

Подобные изменения были внесены как в программный код циклов методов интерполяции (обратных расстояний, кригинг, естественных соседей и др.), так и в циклы методов оптимизации границ карьеров по методу псевдопотока со структурой октодерева (а также плавающего конуса, Лерчса-Гроссмана, метода роя частиц и генетического алгоритма).

Для тестирования была использована блочная модель из стандартной библиотеки моделей MineLib [Espinoza, 2013], предназначенной для решения задач открытой разработки полезных ископаемых. Представленный в библиотеке исходный код pseudo.c, написанный на языке C, использовался для создания версии для распараллеливания. Программная реализация написана на OpenCL 2.1. и предназначена для выполнения на гетерогенных вычислительных системах с многоядерными CPU, оснащенных как внутренними GPU, так и, возможно, имеющих внешние графические адаптеры GPU, подключаемые по интерфейсу Thunderbolt3 NVIDIA. Тесты показали (рис. 4), что метод LLHQ (верхняя метка с очередью по приоритету) является серьезной альтернативой методу LG, поскольку дает в результате такие же по конфигурации контуры конечных карьеров как и метод LG, однако существенно более производительный, особенно на больших блочных моделях.

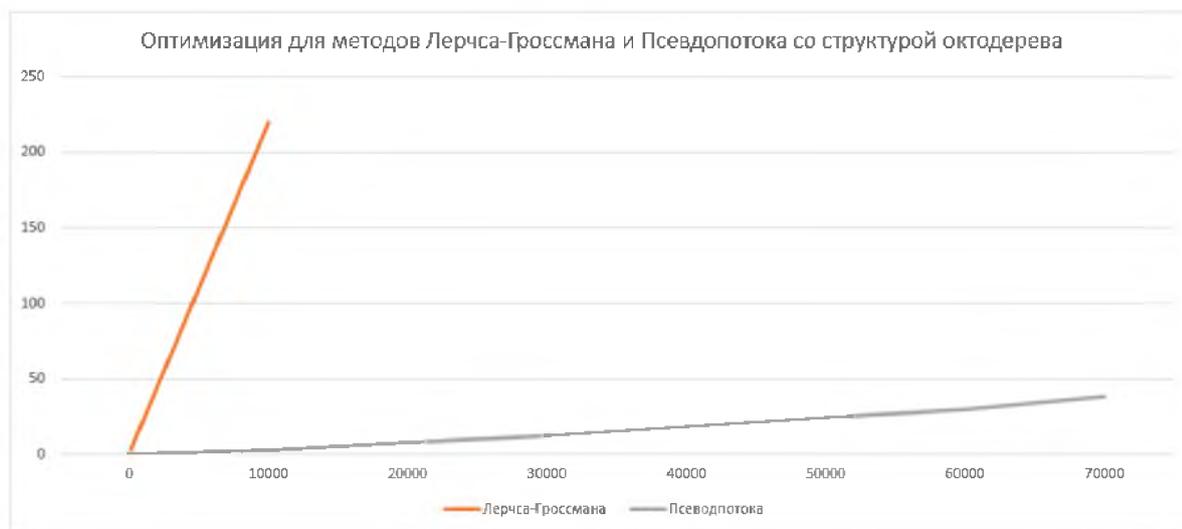


Рис. 4. Сравнение производительности методов оптимизации Лерчсу-Гроссмана и Псевдопотока  
Fig. 4. Compare performance optimization techniques Lerch-Grossman and Pseudo Stream

По модели `zuck_medium` из [Espinoza, 2013] время расчета конечного оптимального контура сократилось примерно на порядок. Для гигаблочных моделей сложных месторождений метод оказывается еще более эффективным и производительным за счет сокращения опций настройки параметров, лучшего распараллеливания потоков извлекаемых блоков и, соответственно, уменьшения требуемого времени обработки. Для ускорения решения задачи поиска оптимального конечного контура оболочки карьера авторами предложен подход на основе распараллеливания вычислений по технологии CUDA/OpenCL при расчетах по методу максимизации псевдопотока с верхней меткой при структуре данных в виде октодеревя на блочных моделях.

Работа выполнена при финансовой поддержке РФФИ в рамках научных проектов  
№ 15-47-03029\_р\_центр\_а и № 16-07-00399\_А

### Список литературы References

- Ахо А.И., Хопкрофт Д., Ульман Д.Д. 2003. Структуры данных и алгоритмы: Перв. с англ., М., Издательский дом "Вильямс", 384 (Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman 2000. Data Structures and Algorithms, Addison-Wesley publishing company, London: 386)
- Aho A.I., Hopcroft D., Ul'man D.D. 2003. Struktury dannyh i algoritmy.: Perv. s angl. [Data Structures and Algorithms], M., Izdatel'skij dom "Vil'jams", 384 (Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman 2000. Data Structures and Algorithms, Addison-Wesley publishing company, London: 386)
- Васильев П. В., Михелев В.М., Петров Д.В. 2015. Оценка вычислительной сложности алгоритмов оптимизации границ карьеров в системе недропользования. Белгород, Издательство БелГУ: 110-120.
- Vasil'ev P. V., Mihelev V.M., Petrov D.V., 2015. Ocenka vychislitel'noj slozhnosti algoritmov optimizacii granic kar'erov v sisteme nedropol'zovanija [Computational complexity for open pit optimization algorithms in mining mineral reserves] Belgorod, Izdatel'stvo BelGU: 110-120. (in Russian)
- Espinoza D, Goycoolea M, Moreno E, Newman A. 2013. Minelib 2011: A library of open pit production scheduling problems. Ann. Oper. Res., 206(1): 93-114.
- Hochbaum D.S. 2008. The Pseudoflow Algorithm: A New Algorithm for the Maximum-Flow Problem. Operations Research, 4: 992-1009.
- Jiang D. 1996. Set operations between linear octrees. Elsevier, Computers & Geosciences, 22/5: 509-516.
- Muir D.C.W. 2008. Pseudoflow, New Life for Lerchs-Grossmann Pit Optimisation. Spectrum Series, Orebody Modelling and Strategic Mine Planning, 14: 97-104.