Экспериментальное сравнение задач и моделей планирования биохимического производства.

В. В. Пырэу, С. А. Тренин
kondratiukvitalik@gmail.com; s.trenin@gmail.com

Целью данной научной работы является комплексное исследование задачи оперативного планирования производства для биохимической промышленности. Исследуются различные постановки задачи составления оптимальных расписаний, учитывающие различные ограничения, приходящие из практики: особенности хранения промежуточных веществ, требования к работе производственных узлов и особенности подготовки станков, такие как наладка и очистка между запусками. Основной класс рассматриваемых математических моделей - смешанное целочисленное линейное программирование, что означает, что рассматриваемые задачи являются NP-трудными. Сложность заключается в том, что схожие задачи могут решаться одинаковыми методами с разной степенью эффективности, что негативно сказывается на процессе внедрения моделей в практику. Для решения этого вопроса проводится экспериментальный запуск моделей, разработанных для одной предметной области, в задачах из другой, интерпретация полученных результатов и предложение эвристик для ускорения алгоритмов.

Ключевые слова: *planning; sheduling; MILP*

DOI:

1 Введение

На сегодняшний день наблюдается высокая конкуренция в разных областях биохимического производства, а так же усложнение производственных процессов, увеличение числа этапов, количества оборудования и объемов продукции. Это неизбежно влечёт появление естественных требований к алгоритмам планирования: они должны быть масштабируемыми, работать за разумное время, находить качественные приближения к оптимальному расписанию и быть гибкими к изменению начальных условий. Основной объект исследования — это различные варианты постановки задачи создания расписания как задачи оптимизации, методы её решения и эвристики, учитывающие индивидуальные особенности задач. На данный момент широкое распространение получили модели смешнанного целочисленного линейного программирования(ЦЛП), так как соответствующие задачи хорошо изучены и существуют алгоритмы для их решения. Однако временные затраты и степень оптимальности найденных решений сильно зависят от количества переменных и ограничений в модели, что делает процесс моделирования значимым для создания плана.

Большинство статей в данной области посвящены конкретным постановкам задач, приходящим из практики и созданию конкретных моделей для их решения. При этом задачи сходны друг другу, хоть и принадлежат разным предметным областям: фармацевтической, пищевой, химической и др. Это, в свою очередь, позволяет применять идеи, высказанные для решения одной задачи к решению другой. Применение имеющихся моделей и эвристик к задачам, для которых они не были разработаны изначально позволит перенять имеющийся опыт, а так же провести тонкую настройку модели под конкретную постановку, что должно привести к улучшению качества.

Некоторые авторы предлагают пути упрощения модели с целью ускорить процесс получения результата без значительной потери его качества. Примером подобной эвристики является двухступенчатая схема, представленная в [1]. В работе проводится анализ других способов упрощения модели и сравнительная оценка результатов.

2 Описание имеющихся данных

Так как авторы статьи поставили одной из своих целью собрать различные варианты постановок задач планирования из разных предметных областей, то в первую очередь необходимо эти задачи описать. Что есть задача автоматического планирования? Пусть описан некоторый процесс производства продукта в виде последовательности операций, которые необходимо произвести с имеющемися прекурсорами.

Определение 1. **Прекурсор** — вещество, имеющееся изначально на складе, из которого будут произведены все требуемые продукты. Также **промежуточным прекурсором** будем называть вещество, получаемое после некоторых этапов производства, но не требуемое для получения в финале.

В описание задачи входят описания всех прекурсоров, описание производственных узлов — сущностей, способных проводить реакции и описания рецептов приготовления промежуточных прекурсоров и финальных продуктов. Рецепт представляет собой описание одной операции: сколько нужно обрабатывать, на каком оборудовании и какие прекурсоры в какой пропорции, чтобы получить продукт.

Вещества разрешается хранить на складе, который описывается максимальной вместимостью. Некоторые вещества, возможно, хранить нельзя вовсе.

В данной работе рассматривается только **пакетное производство**. В этом случае на узел подаётся пакет некоторого размера и обрабатывается. Продуктом является пакет вещества-продукта. Для каждого узла известен максимальный и минимальный размер пакета, который можно подать. Этот подход является альтернативным к **непрерывному производству**, где узлы оперируют с непрерывными потоками данных, но это требует другого подхода к созданию моделей, и поэтому не рассматривается для перекрёстного запуска моделей.

В зависимости от задачи, время обработки пакета может как зависеть, так и не зависеть от его размера. В дальнейшем мы опишем, какие модели способны работать с неравными временами обработки пакета и насколько изменится время получения расписания.

Об оборудовании может быть известно время, необходимое для его настройку в начале производства, выключение в конце и очистку/перенастройку между запусками. О важности этих ограничений написано в [2], где помимо этого присутствуют данные о реальном процессе производства йогуртов. Будет изучено, как добавление этих ограничений влияет на скорость и сложность модели.

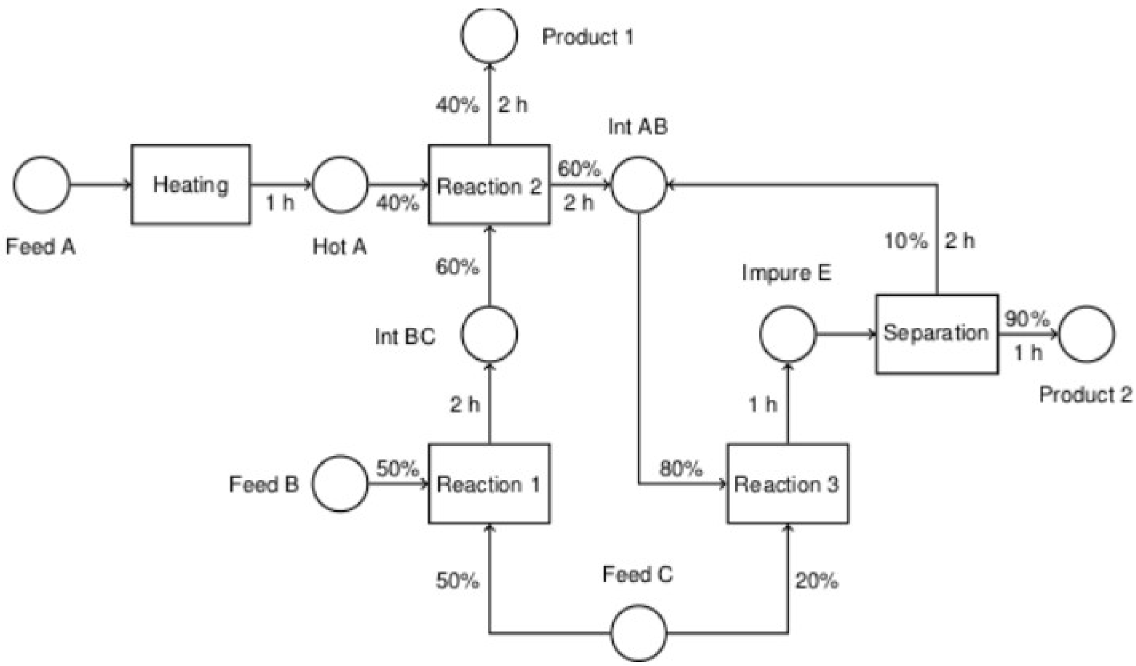
Зададимся вопросом о том, что есть хорошее расписание? Этот вопрос сводится к тому, какие величины входят в целевую функцию. В [3] описаны несколько видов целевой функции: общее время работы системы перед получением требуемых продуктов, разного рода стоимости (стоимости запуска узлов, хранения продуктов и др.), максимизация выгоды от продажи продуктов. Частью поставленного исследования является изучение влияния целевой функции на время получения расписания и его качество.

Процессы, в зависимости от их структуры делятся на две большие группы: последовательные и сетевые.

Определение 2. **Последовательный процесс** — процесс, который может быть разделён на несколько стадий, упорядоченных во времени, и пакеты передаются на производство только от предыдущей стадии к следующей. В зависимости от числа стадий эти процессы делятся на **одноступенчатые** и **многоступенчатые**.

Собственно, **сетевой** процесс — это тот процесс, что не является последовательным. Как будет показано далее, сетевые процессы являются более общими, но и более сложными для моделирования.

Рис. 1 Пример STN некоторого процесса



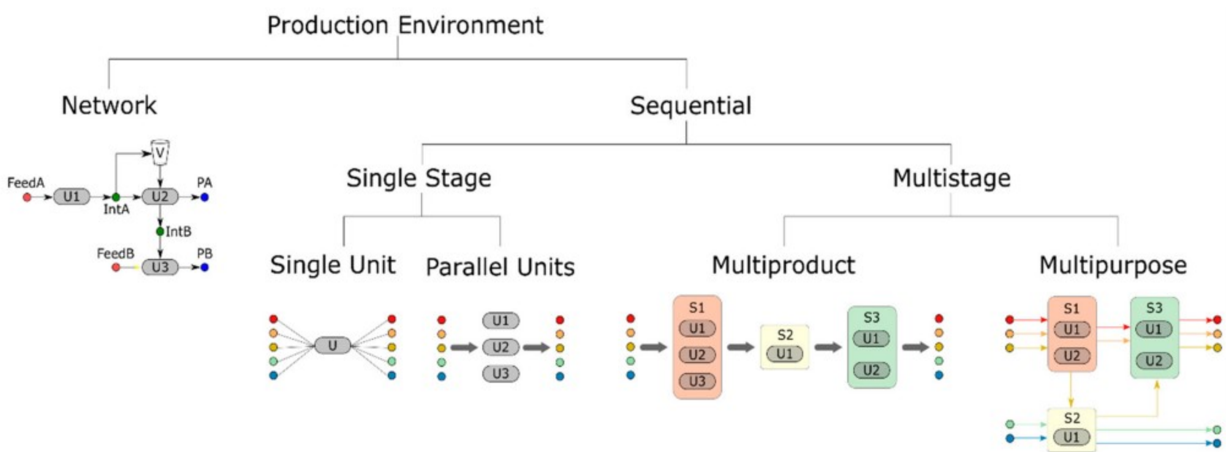
Процессы (в особенности сетевые) удобно представлять себе в виде графов состояний (STN), впервые предложенных в работе [6]. Пример такого графа можно видеть на рисунке 1. Круглыми вершинами обозначаются склады веществ, прямоугольными — задачи, а рёбрами — поток материала (число над ребром — процент вещества в пакете).

Классификация 2 взята из [3] где она является более подробной, но в этой статье параллелизм и наличие многих продуктов/прекурсоров являются параметрами модели. Основной вопрос состоит в том, как переход от последовательных процессов к сетевым изменяет модель и какие методы, изобретённые для последовательных моделей применимы к сетевым и наоборот.

3 Описание моделей

Моделью в рамках данной работы будем называть задачу смешанного целочисленного линейного программирования в общей форме. На данном этапе необходимо закодировать решения, принимаемые системой составления расписаний в переменные, которые будут называться **решающими переменными**. Далее T будет обозначать множество промежутков времени и переменные, с ним связанные. Другими большими латинскими буквами (с индексами) будут обозначаться непрерывные решающие переменные, малыми — бинарные решающие переменные и индексы (в некоторых моделях бинарные переменные и являются своеобразными «индексами» того, что какое-то утверждение верно или нет). Также большими буквами будут обозначаться параметры модели и их множества. Строч-

Рис. 2 Классификация процессов



ные греческие буквы по умолчанию означают разные затраты, сопровождающие процесс (в моделях, где они присутствуют).

Опишем все обозначения, которые будут встречаться в эксперименте.

1. Множества:
   - U — множество производственных узлов. Индексы у этого множества означают:
     - U_j — узлы, способные производить задачу $j \in J$.
   - P — множество продуктов (как прекурсоры, так и те, что получаются в процессе реакций). Индексы у этого множества означают:
     - P_j^{in} — продукты, потребляемые задачей $j \in J$.
     - P_j^{out} — продукты, производимые задачей $j \in J$.
     - $P^{‘}$ — продукты, которые надо произвести к концу (заказ).
   - T — множество временных промежутков
   - J — множество задач (задача есть производство некоторого вещества по его рецепту). Индексы у этого множества означают:
     - J_p^{in} — задачи, потребляющие продукт $p \in P$.
     - J_p^{out} — задачи, производящие продукт $p \in P$.
     - J^u — задачи, которые могут быть произведены на узле $u \in U$.
2. Параметры:
   - B_u^{min}, B_u^{max} — минимальный и максимальный размеры пакета для запуска узла $u \in U$.
   - S_p^{max} — максимальное количество продукта $p \in P$, которое может находиться на складе в любой момент времени.
   - D_p — внешние требования на производство продукта $p \in P^{‘}$.
   - I_p — изначальное количество продукта $p \in P$ на складе.
   - $\mathcal{T}_{u,j}$ — время работы задачи $j \in J$ на узле $u \in U$. По умолчанию считается, что время исполнения задачи не зависит от размера пакета. Будут рассмотренны случаи, в которых время может зависеть от размера пакета, в таком случае этот параметр значит «время работы за килограмм входных веществ».
   - $\mathcal{Q}_{p,j}^{in}, \mathcal{Q}_{p,j}^{out}$ — пропорции входного/выходного продукта $p \in P$ в пакете в рецепте задачи $j \in J$ в случае мультипотребления/мультипроизводства.

- θ_u — стоимость запуска оборудования $u \in U$ за единицу времени.
- ψ_p — стоимость хранения продукта $p \in P$ на складе.
- $\eta_{p,u}$ — стоимость производства продукта $p \in P$ на узле $u \in U$ за килограмм.

В моделях будут рассматриваться две целевые функции:

1.
$$\begin{gathered} min\, MS \\ s.t. \\ MS \leqslant T_{u,i}\ \forall u \in U, i \in \mathbb{N} \end{gathered} \tag{1}$$

Минимизация общего времени выполнения (makespan) при условиях, что оно больше, чем время окончания каждого запуска каждого узла. В зависимости от представления времени в модели величина справа будет по-разному выражаться из переменных.

2.
$$\begin{gathered} min \sum_{p \in P, j \in J_p^{out}, u \in U_j, n \in \mathbb{N}} \eta_{p,u} \mathcal{Q}_{p,j}^{out} B_{j,n} + \sum_{j \in J, u \in U_j, n \in \mathbb{N}} \mathcal{T}_{u,j} * \theta_u * x_{u,j,n} + \sum_{p \in P, t \in T} S_{p,j,t} \\ s.t. \\ S_{p,j,t} = \sum j \in J_p^{out}, Tu, j <= t B_{p,j,n} - \sum j \in J_p^{in}, Tu, j <= t B_{p,j,n} \\ B_u^{min} \leqslant B_{p,j,n} \leqslant B_u^{max} \\ \forall p \in P, j \in J_p^{out}, u \in U_j, n \in \mathbb{N} \end{gathered} \tag{2}$$

Минимизация стоимости производства, в которую входит стоимость работы узлов за время, стоимость производства продуктов за массу и стоимость хранения продуктов. Параметры $B_{j,n}$ — размер пакета, подаваемого на вход n-того запуска задачи j, $x_{u,j,n}$ — индикатор того, что n-тый запуск задачи j совершился на узле u будут считаться в каждой модели по-своему (данные уравнения отражают суть целевых функций и будут отличаться в зависимости от подхода).

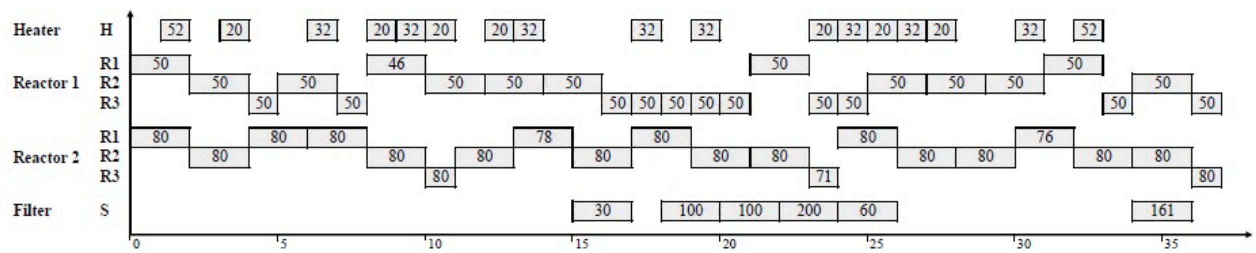
3. Минимизация некоторого общего показателя (например, взвешенной суммы), зависящего от стоимости издержек и временных затрат.

4 Вычислительный эксперимент

Эксперимент состоит в том, что практическая задача будет закодирована разным способом в набор решающих переменных и ограничений и подана на вход COIN-OR Branch-and-Cut (CBC) алгоритму, а далее полученные значения будут интерпретированы и визуализированы в виде диаграмм Ганта. Способ кодирования плана для солвера и интерпретации зависит от подхода к моделированию времени и является основным объектом исследований. Пример диаграммы Ганта можно найти в [5] и на рисунке 3. Эта диаграмма соответствует некоторому плану для процесса 1. По оси Ox отложено время, а по оси oY — производственные узлы. Прямоугольники — это запуски задачи, а числа в них обозначают размер пакета для обработки. Такие диаграммы помогают визуально понять качество расписания и работы решающих алгоритмов.

Основные метрики сравнения подходов к моделированию: количество переменных, количество ограничений, качество расписания, получаемого за ограниченное время, время, необходимое для получения оптимального расписания.

Рис. 3 Пример диаграммы Ганта



5 Модели

5.1 Базовая модель с дискретным временем

Для начала опишем ещё одну классификацию рассматриваемых моделей. Как было описано ранее, от алгоритма ожидаются ответы на два основных вопроса: Когда запускать задачи на узлах и какое количество вещества подавать на вход. Размер пакета кодируется достаточно просто — это переменная непрерывного типа (нет требования на целочисленность). О времени есть два основных подхода, хорошо описанные в [reallife]: моделирование **порядка** запусков на узлах и моделирование фиксированного **времени** начала запуска. Говоря формально, в первом случае вводятся переменные-индикаторы того, что один процесс начинается раньше другого, а потом выбираются времена как непрерывные переменные. Во втором случае временная шкала делится на периоды и вводятся переменные-индикаторы того, что процесс начался в заданный момент. После этого выбираются времена, как непрерывные переменные. Помимо этого, можно классифицировать второй тип дальше: считаем ли мы ширину промежутка фиксированной (между точками на шкале проходит одинаковый период времени) или плавающей (длина промежутка или точное значение момента времени — переменная).

Базовая модель, описываемая в этой части будет принадлежать второму типу: мы поделим заранее шкалу на моменты времени и введём переменные для каждого запуска в каждый момент времени. Эта модель взята из [lpheuristic] и критикуется за большое число переменных в случае плотной временной сетки и низкое качество в случае разряженной.

Опишем переменные детальнее:

1. MS — общее время работы системы.
2. $b_{u,j,t}$ — общее количество вещества, потребляемого процессом $j \in J$ на узле $u \in U$ в момент времени $t \in T$, где T — натуральные числа некоторого отрезка (времена на шкале). Заранее подбирается константа, являющаяся некоторой оценкой на общее время работы.
3. $s_{p,t}$ — количество вещества $p \in P$ на складе в момент времени $t \in T$. Считается, что $s_{p,0}$ даны (начальное состояние складов).
4. $x_{u,j,t}$ — индикатор того, что задача $j \in J$ началась на узле $u \in U$ в момент времени $t \in T$.

Опишем ограничения:

1. $MS \geqslant x_{u,j,t}t + \mathcal{T}_{u,j}\ \forall u \in U, j \in J, t \in T$ — общее время работы не меньше, чем время окончания каждого процесса.

2. $x_{u,j,t}B_u^{min} \leqslant b_{u,j,t} \leqslant x_{u,j,t}B_u^{max}\ \forall u \in U, j \in J, t \in T$ — пакет, потребляемый процессом $j \in J$ на узле $u \in U$ лежит между максимальным и минимальным размерами, которые узел может принять. Если процесс не запускается, то вещества он не потребляет.
3. $s_{p,t} = s_{p,t-1} + \sum\limits_{j\in J_p^{out}, u\in U_j, t-\mathcal{T}_{u,j}\geqslant 1} \mathcal{Q}_{p,j}^{out} b_{u,j,t} - \sum\limits_{j\in J_p^{in}, u\in U_j, t\geqslant 1} \mathcal{Q}_{p,j}^{in} b_{u,j,t}\ \forall t \geqslant 1, p \in P$ — уравнения баланса склада. Количество вещества в момент времени t есть количество вещества в момент $t-1$ плюс то, что успели к этому моменту произвести, минус количество, которое потребляют начатые процессы.
4. $0 \leqslant s_{p,t} \leqslant S_p^{max}$ — ограничения на объем хранящихся на складе веществ.
5. $\sum\limits_{j\in U_j, t'\in[t-\mathcal{T}_{u,j},t] x_{u,j,t'}} \leqslant 1\ \forall t \in T, u \in U$ — ограничения одновременности. В момент времени t узел u выполняет не более одной задачи.
6. $s_{p,0} = I_p\ \forall p \in P$ — начальное состояние складов.
7. $s_{p,max(T)} = D_p\ \forall p \in P$ — требование на заказ.

При использовании этого подхода к процессу, изображенному на диаграмме 1 и данных, описанных в [5] было получено 20860 решающих переменных. За 10 минут работы было найдено расписание, затрачивающее 60 рабочих часов, что уступает почти в 2 раза результатам, полученным в оригинальной работе [5].

6 Заключение

Желательно, чтобы этот раздел был, причём он не должен дословно повторять аннотацию. Обычно здесь отмечают, каких результатов удалось добиться, какие проблемы остались открытыми.

Литература

[1] *F. Blomer, H.-O. Gunther* LP-based heuristics for scheduling chemical batch processes, 2010 International Journal of Production Research, 38:5, 1029-1051 doi: `http://dx.doi.org/10.1080/002075400189004`.

[2] *Georgios P. Georgiadis, Georgios M. Kopanos, Antonis Karkaris, Harris Ksafopoulos and Michael C. Georgiadis* Optimal Production Scheduling in the Dairy Industries, 2019 Industrial & Engineering Chemistry Research 58 (16), 6537-6550 doi: `http://dx.doi.org/10.1021/acs.iecr.8b05710`.

[3] *Georgiadis, Georgios P. and Elekidis, Apostolos P. and Georgiadis, Michael C.* Optimization-Based Scheduling for the Process Industries: From Theory to Real-Life Industrial Applications, 2019 Industrial & Engineering Chemistry Research 58 (16), 6537-6550 doi: `http://dx.doi.org/10.3390/pr7070438`.

[4] *Siqun Wang, Monique Guignard* Hybridizing Discrete- and Continuous-Time Models For Batch Sizing and Scheduling Problems, 2006 Computers & Operations Research Volume 33, Issue 4 doi: `http://dx.doi.org/10.1016/j.cor.2004.11.013`.

[5] *Christos T. Maravelias and Ignacio E. Grossmann* Minimization of the Makespan with a Discrete-Time State-Task Network Formulation, 2003 Industrial & Engineering Chemistry Research doi: `http://dx.doi.org/10.1021/ie034053b`.

[6] *E.Kondili, C.C.Pantelides, R.W.H.Sargent* A general algorithm for short-term scheduling of batch operations—I. MILP formulation, 1993 Computers & Chemical Engineering doi: `http://dx.doi.org/10.1021/ie034053b10.1016/0098-1354(93)80015-F`.

Поступила в редакцию