

© 2017 г. А.Б. ДОЛГИЙ, д-р техн. наук
(alexandre.dolgui@mines-nantes.fr)
(Горная школа Нанта, Нант, Франция),
А.В. ЕРЕМЕЕВ, д-р физ.-мат. наук (eremeev@ofm.oscsbras.ru)
(Институт математики им. С.Л. Соболева СО РАН, Новосибирск),
В.С. СИГАЕВ (sigvs@mail.ru)
(ООО «Автоматика-сервис», Омск)

АНАЛИЗ ЗАДАЧИ МНОГОКРИТЕРИАЛЬНОЙ ОПТИМИЗАЦИИ ЕМКОСТИ БУНКЕРОВ В ПРОИЗВОДСТВЕННОЙ ЛИНИИ¹

Рассматривается задача многокритериальной оптимизации объемов бункеров в производственной линии. Предполагается, что линия имеет параллельно-последовательную структуру и в процессе работы происходят случайные по моменту возникновения и длительности останковки оборудования, вызванные его отказами. Объемы бункеров целочисленны и ограничены сверху. В качестве критериев рассматриваются средняя производительность линии, капитальные затраты на установку бункеров и стоимость складирования промежуточной продукции. Для аппроксимации парето-оптимального множества используются эволюционные алгоритмы SIBEA и SEMO. На задачах большей размерности экспериментально показано преимущество модифицированного алгоритма SEMO по значению гиперобъема полученного множества точек.

Ключевые слова: производственная система, эволюционные алгоритмы, производительность, капитальные затраты, стоимость складирования

1. Введение

При создании автоматизированных систем управления производством и ряда других систем поддержки принятия решений большое значение имеет разработка алгоритмов аппроксимации множества парето-оптимальных решений. При управлении такими производственными системами, как автоматические линии, гибкие производственные системы или автоматизированные сборочные линии, в которых детали перемещаются от одного станка к другому с помощью некоторого транспортного механизма, возникает следующая задача оптимизации объемов бункеров.

Вследствие отказов оборудования в процессе работы линии возникают останковки единиц оборудования (ЕО), случайные по моменту возникновения и длительности.

¹Работа выполнена при финансовой поддержке проекта Российского научного фонда (грант № 15-11-10009). Это препринт Материалов, принятых для публикации в журнал Автоматика и телемеханика (С) (2017) Pleiades Publishing, Ltd. Издатель: <http://www.maik.ru>.

Последствия отказов распространяются на смежные операции из-за невозможности передать деталь на следующую операцию или отсутствия деталей на входе ЕО. Наличие бункеров (емкостей) для складирования деталей между ЕО позволяет снизить влияние отказов на соседние операции и повысить среднюю производительность линии. Однако установка бункеров связана с дополнительными капитальными затратами и увеличением числа складироваемых деталей. Задача состоит в выборе объемов бункеров с учетом средней производительности линии, капитальных затрат на установку бункеров и стоимости хранения деталей.

Значимость решения подобных задач оптимизации производственных линий продемонстрирована в [1]. Существенный экономический эффект от внедрения методов решения таких задач на автозаводах PSA Peugeot Citroën показан в [2].

Анализ производственных линий с учетом отказов оборудования, как правило, ведется посредством построения марковских моделей с дискретным или непрерывным временем в предположении геометрических или экспоненциальных распределений времени безотказной работы ЕО и времени их восстановления (см., например, [3]). Длительность обработки детали может предполагаться детерминированной или случайной (как правило с геометрическим, экспоненциальным или эрланговым распределением вероятностей). В случае непрерывного времени и детерминированных длительностей обработки деталей отдельные немарковские переходы аппроксимируются марковскими в предположении экспоненциального распределения соответствующих случайных величин [4–6]. При достаточно естественных предположениях полученные таким образом марковские модели имеют стационарное распределение (см. [7], гл. 2) и для них определены средняя производительность и среднее число деталей в каждом бункере в стационарном режиме.

Большая часть работ в литературе по задаче оптимизации объемов бункеров связана с однокритериальными постановками (см., например, [8–10]). В других работах рассматривалось более одного критерия, но с использованием их взвешенной суммы [11, 12]. В [13] алгоритм муравьиных колоний и эволюционный алгоритм из [14] адаптированы для многокритериальной задачи оптимизации объемов бункеров. При этом в качестве критериев рассматривались максимизация средней производительности линии, рассчитанная имитационным алгоритмом, и минимизация общего размера бункеров. Известный вариант многокритериального генетического алгоритма [15] в работе [16] адаптирован к двухкритериальной задаче распределения бункеров, где в качестве критериев выступают средняя производительность линии в стационарном режиме и капитальные затраты на установку бункеров.

Отличительной особенностью настоящей статьи является то, что в постановке задачи используется три критерия: максимизация средней производительности линии в стационарном режиме, минимизация капитальных затрат на установку бункеров и минимизация средней стоимости складирования деталей в промежуточных бункерах. При расчете средней производительности линии и среднего числа деталей в бункерах используется эффективный приближенный метод её анализа [4], основанный на замене участков линии «эквивалентными» ЕО.

В отличие от однокритериальных вариантов задачи, где используется свертка критериев или часть критериев заносится в ограничения, задача в трехкритериальной постановке имеет не одно оптимальное значение критерия, а целое множество неуплучшаемых значений вектора из трех критериев (фронт Парето). Рассматривае-

мая трехкритериальная постановка задачи представляется наиболее актуальной на этапе поиска пути развития производства, когда еще не выбрана плановая производительность и производственная база. На этом этапе анализ фронта Парето позволяет лицу, принимающему решения (ЛПР), сделать свой выбор среди множества неулучшаемых вариантов распределения бункеров. На более поздних этапах принятия решений, когда задана требуемая производительность или ограничен суммарный объем бункеров, могут быть более адекватны однокритериальные постановки задачи.

Точные методы вычисления средней производительности линии известны в случаях двух и в некоторых частных случаях для трех последовательных ЕО, разделенных бункерами (см., например, обзор [3]). Для общего случая применяются приближенные методы декомпозиции, агрегирования или имитационного моделирования [3, 4].

В связи с отсутствием точных методов вычисления средней производительности линии и среднего числа складированных деталей в настоящей работе для аппроксимации множества Парето используются эволюционные алгоритмы, применение которых для этой задачи не требует точного вычисления критериев.

2. Постановка задачи

Пусть n – количество бункеров, m – число ЕО в линии. Через \mathbb{Z}_+ обозначим множество неотрицательных целых чисел. Введем вектор переменных $\mathbf{h} = (h_1, \dots, h_n) \in \mathbb{Z}_+^n$, где h_i – объем i -го бункера, $i = 1, \dots, n$. Структура линии представляется ориентированным последовательно-параллельным графом G , у которого вершины b_0, \dots, b_{n+1} отвечают бункерам, а дуги a_1, \dots, a_m – единицам оборудования. Каждая дуга направлена от вершины входного бункера соответствующей ЕО к вершине его выходного бункера. Возможны кратные дуги, т.е. параллельно работающие ЕО с общими бункерами. Единственная вершина b_0 , не имеющая входящих дуг, соответствует входному бункеру линии, а единственная вершина b_{n+1} без выходящих дуг – выходному. Каждой вершине b_j приписано максимальное число вмещаемых деталей h_j в бункер j , причем $h_0 = h_{n+1} = \infty$. Предположим, что на входе линии всегда имеется достаточное число деталей и обработанные детали всегда могут быть помещены в выходной бункер, поскольку он имеет неограниченный объем.

Любая ЕО может находиться в состояниях «работа», «отказ» (идет восстановление ЕО после ее поломки), «блокировка» (невозможно передать обработанную деталь на следующую операцию) и «простой» (отсутствие деталей на входе ЕО). Согласно сделанным выше предположениям ЕО, связанные с бункером на входе системы, не могут находиться в состоянии «простой», а ЕО, связанные бункером на выходе системы, не могут находиться в состоянии «блокировка».

В случае заполненности бункера на выходе некоторой ЕО, обработанная ею деталь не может быть убрана с ЕО и остается в ЕО до момента появления свободного места в бункере. В ситуации, когда несколько ЕО ожидают свободного места в одном и том же бункере, в момент освобождения места для одной детали разблокируется одна из этих ЕО, выбранная равновероятно. В ситуации, когда несколько ЕО ожидают поступления деталей из некоторого в настоящее время пустого бункера, в момент появления детали в бункере из простоя выходит одна из этих ЕО, выбранная равновероятно (появившаяся деталь загружается на эту ЕО для обработки).

Каждая дуга a_i графа G характеризуется тройкой параметров $(T_i^O, T_i^B, U_i) \in$

$[1, \infty)^2 \times \mathbb{N}$, где \mathbb{N} – множество натуральных чисел. Находясь в состоянии «работа», ЕО с номером i имеет постоянное время обработки детали $U_i \in \mathbb{N}$, $i = 1, \dots, m$. Отказы и восстановления различных ЕО происходят независимо, и время наработки на отказ, также как и время восстановления ЕО, имеет геометрическое распределение. Для каждой ЕО $T_i^O \in [1, \infty)$ – среднее время наработки на отказ, а $T_i^B \in [1, \infty)$ – среднее время восстановления. Если ЕО с номером i исправна, а ее входной бункер j не пуст и бункер j' на ее выходе не полон (т.е. она не находится в состоянии «отказ», «блокировка» или «простой»), то каждые U_i единиц времени данная ЕО пополняет бункер j' одной деталью и извлекает одну деталь из бункера j . В состояниях «блокировка» и «простой» ЕО не отказывает. В состояниях «отказ», «блокировка» и «простой» ЕО не вносит изменений в заполненность ее входного и выходного бункеров. В момент отказа обработка текущей детали прерывается, а после восстановления продолжается до завершения. Время обработки детали после восстановления равно оставшемуся времени обработки в момент отказа.

Возможность моделировать рассматриваемую производственную линию цепью Маркова с дискретным временем обсуждается в Приложении 1. Там же дается формальное определение средней производительности линии $V(\mathbf{h})$ и среднего числа деталей $q_j(\mathbf{h})$ в бункере j , $j = 1, \dots, n$, в стационарном режиме.

Определим множество допустимых решений рассматриваемой задачи выбора размеров бункеров как $D = \{\mathbf{h} | 0 \leq h_i \leq d_i, i = 1, \dots, n\}$, где d_i – максимально допустимый размер бункера i , $i = 1, \dots, n$. В данной работе рассматривается многокритериальная задача оптимизации объемов бункеров для последовательно-параллельной структуры линии со следующими критериями:

- средняя производительность линии в стационарном режиме $V(\mathbf{h}) \rightarrow \max$;
- капитальные затраты на установку бункеров $J(\mathbf{h}) \rightarrow \min$, где $J(\mathbf{h}) = \sum_{j=1}^n J_j(h_j)$, $J_j(h_j)$ – стоимость установки бункера j объемом h_j ;
- стоимость складирования деталей в промежуточных бункерах $Q(\mathbf{h}) \rightarrow \min$, где $Q(\mathbf{h}) = \sum_{j=1}^n w_j q_j(\mathbf{h})$, w_j – стоимость складирования деталей в бункере j .

2.1. Схемы агрегирования участков линии

По аналогии с [4–6, 17, 18] и некоторыми другими работами далее предполагается, что количество деталей x_j в бункере j измеряется не целым числом, а непрерывной величиной из $[0, h_j]$. Это упрощение связано с необходимостью сократить вычислительные затраты и основано на наблюдении, что только граничные ситуации, когда заполненность бункера равна 0 или h_j , могут привести к остановке оборудования. При всех остальных состояниях производительность линии не изменяется, каково бы ни было состояние бункера в $(0, h_j)$. Переход от дискретных состояний к непрерывным позволяет заменить одним состоянием с вещественным параметром x_j все множество внутренних состояний с заполненностью бункера в интервале $(0, h_j)$.

В данной работе в качестве алгоритма приближенного вычисления параметров линии в стационарном режиме используется алгоритм из [4], основанный на замене участков линии «эквивалентными» ЕО. Далее приближенные значения для $V(\mathbf{h})$, $q_1(\mathbf{h})$, $q_2(\mathbf{h})$, \dots , $q_n(\mathbf{h})$ и $Q(\mathbf{h})$, найденные с помощью указанного метода агрегирования, обозначаются соответственно через $V'(\mathbf{h})$, $q'_1(\mathbf{h})$, $q'_2(\mathbf{h})$, \dots , $q'_n(\mathbf{h})$ и $Q'(\mathbf{h})$.

Процедуры агрегирования разработаны для двух типов участков линии: с двумя последовательными ЕО (правило R_1) и с двумя параллельными ЕО (правило R_2).

В процессе работы алгоритма последовательно выбираются двухмашинные участки и заменяются одной ЕО с «эквивалентными» параметрами, пока не будет получена линия из одной ЕО. Средняя производительность данной ЕО дает величину $V'(\mathbf{h})$, а средние значения числа деталей в бункерах, исключаемых при агрегировании по правилу R_2 , дают соответствующие величины $q'_j(\mathbf{h})$.

Точность описанной эвристики зависит от порядка агрегирования участков линии. Эффективность различных правил выбора пары ЕО для применения процедуры агрегирования детально исследовалась в [19]. В данном случае правило R_1 всегда, где это возможно, имеет приоритет перед применением правила R_2 . В случае, если имеется несколько альтернатив при выборе пары с последовательными ЕО, приоритетной является пара с бункером наименьшей емкости.

В вычислительных экспериментах [4] на примерах из [18] используемый метод агрегирования показал отклонение в пределах 5% от средних значений, полученных в имитационном моделировании. Вычислением доверительных интервалов для $V(\mathbf{h})$ на некоторых из этих примеров было установлено, что величина $V'(\mathbf{h})$, найденная методом агрегирования, лежит в интервале с доверительным уровнем 0,99.

2.2. Алгоритмы аппроксимации фронта Парето

Пусть на множестве допустимых решений D задана критериальная вектор-функция $\mathbf{f} = (f_1, \dots, f_k)$ со значениями в пространстве критериев $\mathbf{f}(x) = (f_1(x), \dots, f_k(x)) \in \mathbb{R}^k, x \in D$, где k – число критериев. Введем отношение доминирования по Парето в пространстве \mathbb{R}^k для случая, когда имеются все критерии «на максимум»: вектор $\mathbf{f} = \mathbf{f}(x), x \in D$ доминирует по Парето вектор $\bar{\mathbf{f}} = \mathbf{f}(\bar{x}), \bar{x} \in D$, если среди неравенств $f_i(x) \geq f_i(\bar{x}), i = 1, \dots, k$, имеется хотя бы одно строгое. В случае, если имеются критерии «на минимум», отношение доминирования в пространстве критериев вводится аналогично. Решение $x \in D$ доминирует решение $\bar{x} \in D$, если вектор $\mathbf{f}(x)$ доминирует по Парето вектор $\mathbf{f}(\bar{x})$. Множество \tilde{D} всех недоминируемых допустимых решений называется множеством парето-оптимальных решений (паретовским множеством). Фронт Парето называется множеством $F := \mathbf{f}(\tilde{D})$.

Поскольку в общем случае точное вычисление критериев $V(\mathbf{h})$ и $Q(\mathbf{h})$ не представляется возможным, в качестве первого шага аппроксимации фронта Парето воспользуемся заменой отношения доминирования, определяемого критериями V, J и Q , на отношение доминирования, определяемое критериями V', J и Q' . Последнее отношение обозначим через \succ' , а соответствующий ему фронт Парето – через F' . Далее, как следует из результатов [20], точное вычисление F' представляет собой NP-трудную задачу и может потребовать чрезмерных вычислительных затрат. В связи с этим делается второй шаг аппроксимации, состоящий в поиске приближенного решения задачи поиска фронта Парето F' посредством эволюционных алгоритмов SIBEA [21] и SEMO [22].

В процессе работы алгоритма SEMO (Simple Evolutionary Multiobjective Optimizer) на каждой итерации из популяции, которая содержит попарно недоминируемые решения (особи), случайным образом выбирается родительская особь. Далее из родительской особи посредством операции мутации получаем потомка, которого добавляем в популяцию, если в ней нет доминирующих его особей или особей с тем же значением векторного критерия. Все особи популяции, которые доминируются потомком, удаляются. Результатом работы алгоритма является вычисленная к

концу его выполнения популяция Π .

Алгоритм 1 (SEMO).

1. Сгенерировать случайным образом решение x и положить $\Pi := \{x\}$.
2. Пока не выполнен критерий останова:
 - 2.1. Выбрать случайным образом особь x из популяции Π .
 - 2.2. Создать потомка $x' := Mut(x)$ (оператор $Mut(x)$ будет описан позднее).
 - 2.3. Удалить доминируемые особи из популяции, $\Pi := \Pi \setminus \{z \in \Pi \mid x' \succ' z\}$.
 - 2.4. Если $\nexists z \in \Pi$, такой что $z \succ' x'$ или $\mathbf{f}(z) = \mathbf{f}(x')$, то $\Pi := \Pi \cup \{x'\}$.

Критерий останова алгоритма определяется одним из следующих способов:

- остановить, если достигнуто максимальное число итераций;
- остановить, если достигнуто максимальное время, отведенное для запуска.

Особь, выбранная случайным образом из популяции, подвергается действию оператора мутации, при котором выбранное решение изменяется случайным образом. Применительно к рассматриваемой задаче действие оператора мутации определяется функцией $Mut(\mathbf{h}) = (h_1 + \xi_1, \dots, h_N + \xi_N)$, где ξ_i – целочисленные случайные величины, равномерно распределенные от $\max\{-h_j, -\Delta\}$ до $\min\{d_j - h_j, \Delta\}$, $j = 1, \dots, N$. Здесь Δ – параметр алгоритма, задающий интенсивность мутации.

Для описания следующего алгоритма потребуется определение *гиперобъема* [23, 24]. Выберем в пространстве критериев *референтную точку* $\mathbf{r} = (r_1, \dots, r_k)$ (как правило выбирается точка \mathbf{r} , заведомо доминируемая всеми точками $\mathbf{f}(x)$, $x \in D$). Для множества $A \subseteq D$ гиперобъем $I_H(A)$ задается следующим образом:

$$I_H(A) = VOL\left(\bigcup_{a \in A} [r_1, f_1(a)] \times \dots \times [r_k, f_k(a)]\right),$$

где через $VOL(\cdot)$ обозначена мера по Лебегу. В случае, если имеются критерии «на минимум», определение гиперобъема модифицируются очевидным образом.

Алгоритм $(\mu + 1)$ -SIBEA основан на Simple Indicator-Based Evolutionary Algorithm (SIBEA) предложенном в [24]. В процессе работы данного алгоритма на каждой итерации в популяцию Π численности μ добавляется потомок x , созданный из родительской особи, выбранной случайным образом. После чего из популяции удаляется особь, вклад которой в значение гиперобъема популяции по критериям V' , J и Q' является минимальным. В качестве референтной точки была выбрана точка $\mathbf{r} = \left(0, J(d_1, \dots, d_n), \sum_{j=1}^n w_j d_j\right)$. Критерий останова алгоритма определяется такими же способами, как в алгоритме SEMO.

Алгоритм 2 (Схема $(\mu + 1)$ -SIBEA).

1. Сгенерировать случайным образом популяцию Π численности μ .
2. Пока не выполнен критерий останова:
 - 2.1. Выбрать случайным образом особь x из популяции Π .
 - 2.2. Создать потомка $x' := Mut(x)$ и добавить в популяцию $\Pi' := \Pi \cup \{x'\}$ (оператор $Mut(x)$ рассмотрен ранее).
 - 2.3. Определить $d(x) := I_H(\Pi') - I_H(\Pi' \setminus \{x\})$ для каждой особи $x \in \Pi'$.
 - 2.4. Выбрать особь $z \in \Pi'$, такую что $d(z) = \min_{x \in \Pi'} d(x)$. В случае, если имеется несколько таких особей, выбрать случайным образом одну из них.

2.5. Удалить выбранную особь из популяции $\Pi := \Pi \setminus \{z\}$.

Как и у алгоритма SEMO, значение гиперобъема всей популяции не уменьшается с ростом количества итераций, однако в отличие от SEMO размер популяции алгоритма SIBEA остается постоянным.

Алгоритм SIBEA нацелен на аппроксимацию паретовского множества с помощью ограниченного числа точек. Свойство ограниченности численности популяции играет важную роль, если необходимо аппроксимировать парето-оптимальное множество сравнительно небольшим количеством решений.

3. Вычислительный эксперимент

SEMO и SIBEA были реализованы в среде программирования Visual Studio 2010. С целью увеличения разнообразия популяции авторами также рассмотрена модификация алгоритма SEMO, в которой начальная популяция состоит из 1000 особей, сгенерированных случайным образом. Далее данный алгоритм обозначается через SEMO+. Все испытания проводились на ЭВМ Intel Core i5 (процессор 2,4 МГц, ОЗУ 4 Гб). Интенсивность мутации $\Delta = 2$.

В экспериментах использовались серия задач AS из [18] и задача малой размерности P_0 . Задача P_0 создана на основе линии последовательной структуры с тремя промежуточными бункерами и четырьмя ЕО, где $d_1 = d_2 = d_3 = 20$, $T_1^O = T_4^O = 30$, $T_2^O = T_3^O = 15$, $T_1^B = T_4^B = 200$, $T_2^B = T_3^B = 100$, $U_1 = U_2 = U_3 = 3000$. Серия AS состоит из задач, созданных на основе поточных линий № 1,2,6,7,8 из [18] с реальными данными с производства Renault. Тестовый пример № 3 из [18] не использовался, так как он демонстрирует особый случай, когда приближенное вычисление производительности линии V' существенно отличается от результатов имитационного моделирования. Кроме того, как отмечено в [19], примеры такого вида не являются реалистичными.

Линии № 4,5 из [18] не используются ввиду малой размерности. Параметры остальных линий приведены в Приложении 3. Стоимости складирования деталей w_j , $j = 1, \dots, n$, полагались единичными во всех задачах, $J_j(h_j) \equiv h_j$, $j = 1, \dots, n$.

Небольшой размер пространства решений задачи P_0 позволил полностью просмотреть его алгоритмом лексикографического перебора. На основе полученных результатов была проведена оценка скорости нахождения решений из фронта Парето F' в зависимости от количества сгенерированных пробных точек алгоритмами SEMO и SIBEA (под пробной точкой понимается решение, для которого вычисляются значения критериев). В экспериментах время работы алгоритмов было ограничено одним часом.

На рис. 1 продемонстрирован процесс работы алгоритмов, где по оси ординат отображено количество найденных элементов из фронта Парето, а по оси абсцисс – количество сгенерированных пробных точек. Из рис. 1 видно превосходство алгоритма SEMO над SIBEA по количеству найденных элементов фронта Парето F' . Подобная картина наблюдается и на задаче AS1, где пространство решений также удалось полностью исследовать алгоритмом лексикографического перебора. Кроме того, рис. 1 иллюстрирует ситуацию, когда на задаче малой размерности предпочтительнее использовать лексикографический перебор, а не эволюционные алгоритмы.

На задачах с большой мощностью пространства решений, таких как AS2, AS6 и AS7, полный перебор решений за приемлемое время не представляется возможным.

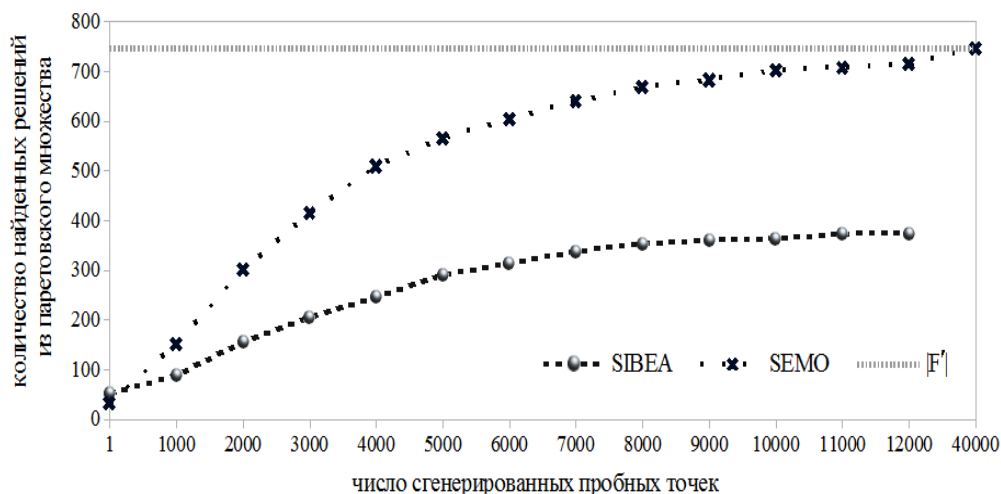


Рис. 1. Число найденных точек из F' в процессе работы алгоритмов. Задача P_0 .

Как уже упоминалось выше, для сравнения алгоритмов на серии задач AS каждому из алгоритмов был предоставлен 1 час работы, по окончании которого рассчитывался гиперобъем полученной популяции. Размер начальной популяции алгоритма SIBEA был равен количеству недоминируемых решений, найденных алгоритмом лексикографического перебора (возможно неполного) за то же время, равное 1 часу. За 1 час счета полный перебор пространства решений был завершен и дал все множество F' только в случае задачи AS1. Перебор велся по возрастанию отношения лексикографического порядка \succ_L , где $\mathbf{h} \succ_L \mathbf{h}'$, если при некотором k выполняется $h_k > h'_k$, $h_{k+1} = h'_{k+1}, \dots, h_n = h'_n$. На рис. 2 продемонстрированы результаты работы алгоритмов на всей серии задач AS. Число решений, просмотренных при лексикографическом переборе, мощность множества D , а также число пробных решений, вычисленных в алгоритмах SIBEA, SEMO и SEMO+, представлены в табл. 1. По оси ординат на рис. 2 отложено отношение гиперобъема популяции, полученной каждым алгоритмом, к лучшему найденному в экспериментах за 1 час значению гиперобъема на данной задаче. На графике видно, что задача AS1 оказалась простой для всех алгоритмов; на задачах AS2 и AS8 значение гиперобъема популяции, полученной SIBEA больше, чем у SEMO; на задачах AS6 и AS7 превосходство имеет алгоритм SEMO.

Было установлено, что на задачах AS2 и AS8 популяция алгоритма SEMO локализуется в сравнительно небольшом подмножестве пространства решений. Для выяснения причины локализации с помощью полного лексикографического перебора было получено все множество F' для задачи AS8 (что заняло более шести часов). Было установлено, что при точности сравнения величин $V'(\mathbf{h})$ и $Q'(\mathbf{h})$ порядка 10^{-15} прообразом фронта Парето F' является множество решений вида $(0, 0, 0, 0, 0, h_6, 0)$, где $h_6 = 0, \dots, 2000$.

На рис. 3 слева отображены значения бункеров 2 и 6 для особей из популяции, по-

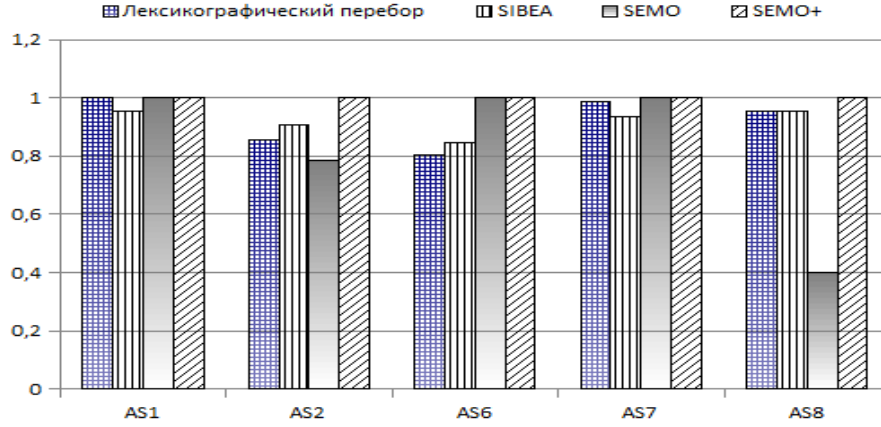


Рис. 2. Относительная величина гиперобъема.

Таблица 1. Размерности задач и число вычисленных пробных решений за 1 час.

задача	n	$ D $	лексикограф. перебор	SIBEA	SEMO	SEMO+
AS1	4	$4 \cdot 10^5$	$1,3 \cdot 10^5$	$1,4 \cdot 10^3$	$8,6 \cdot 10^6$	$8,7 \cdot 10^6$
AS2	9	$9 \cdot 10^{11}$	$1,3 \cdot 10^5$	$2,1 \cdot 10^3$	$1,4 \cdot 10^6$	$9,6 \cdot 10^5$
AS6	13	$2 \cdot 10^{22}$	$1,3 \cdot 10^5$	$1,2 \cdot 10^4$	$8,8 \cdot 10^6$	$7,8 \cdot 10^6$
AS7	7	$9 \cdot 10^7$	$1,3 \cdot 10^5$	$2,2 \cdot 10^4$	$1,7 \cdot 10^7$	$1,6 \cdot 10^7$
AS8	7	$5 \cdot 10^8$	$1,3 \cdot 10^5$	$7,7 \cdot 10^3$	$1,2 \cdot 10^7$	$1,5 \cdot 10^7$

строенной алгоритмом SEMO. Из рисунка видно, что в популяции алгоритма SEMO недостаточно разнообразия для получения всего множества F' . Как показал эксперимент с SEMO+, популяция этого алгоритма не локализовывалась в (см. рис. 3 справа), как в случае SEMO, что позволило более точно аппроксимировать фронт Парето F' .

Для статистического сравнения работы алгоритмов SEMO и SEMO+ на задаче AS8 было произведено по 30 запусков каждого из них до получения всего фронта Парето F' . Среднее количество итераций, необходимое алгоритму SEMO+ для получения всего множества F' , составило $3 \cdot 10^5$, а в случае алгоритма SEMO эта величина была равна $3 \cdot 10^6$. По результатам проведенных экспериментов были сформированы две 30-элементные выборки, где регистрировалось число пробных решений до получения всего множества F' на каждом из 30 запусков алгоритма. Непараметрический критерий Манна-Уитни, примененный для сравнения этих выборок, показал статистическую значимость различий с уровнем $p < 0,01$.

Таким образом, на задаче AS8 алгоритм SEMO+ находит фронт Парето F' быстрее, чем алгоритм SEMO. В целом на всей серии задач AS значения гиперобъема популяций, полученных алгоритмом SEMO+, также оказались наибольшими, как это видно из рис. 2.

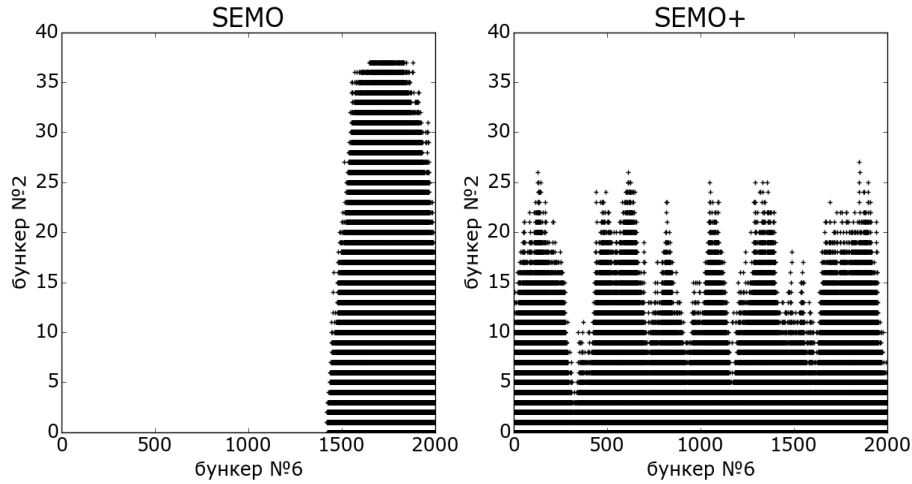


Рис. 3. Полученные популяции алгоритмов SEMO и SEMO+ для задачи AS6.

4. Заключение

Как показали проведенные эксперименты, если число точек в аппроксимации фронта Парето не ограничено сверху некоторой априори выбранной величиной, то алгоритм SEMO+ имеет преимущество перед алгоритмами SEMO и SIBEA по гиперобъему получаемого множества решений. Однако если аппроксимирующее множество должно состоять из достаточно малого числа решений, то алгоритмы SEMO и SEMO+ уступают алгоритму SIBEA.

На практике ЛПР имеет возможность проанализировать лишь сравнительно малое число «перспективных» вариантов распределения бункеров, поэтому наиболее приемлемым для ЛПР представляется использование множества решений, найденных алгоритмом SIBEA. Точность аппроксимации фронта Парето в таком случае может быть приближенно оценена из сравнения гиперобъемов популяций, построенных SIBEA и SEMO+.

Авторы благодарны В.А. Топчию и рецензентам за ряд ценных советов при подготовке статьи.

ПРИЛОЖЕНИЕ 1

Настоящее Приложение содержит марковскую модель производственной линии с параллельно-последовательной структурой, описанной в разделе 2.

Состояния рассматриваемой линии описываются набором параметров $(\alpha_1, \dots, \alpha_m, x_1, \dots, x_n, y_1, \dots, y_m)$. Здесь $\alpha_i = 0$, если ЕО с номером i находится в состоянии «отказ», и $\alpha_i = 1$ иначе. Значение $x_j \in \{0, 1, \dots, h_j\}$ определяет количество деталей в бункере j . Величина $y_i \in \{0, 1, \dots, U_i\}$ – число единиц времени, уже затраченных на обработку имеющейся детали на ЕО с номером i . Значение y_i равно нулю, если ЕО находится в состоянии «простой». Когда ЕО находится в состоянии «блокировка», значение y_i не меняется и остается равным U_i . Множество всех состояний есть $S = \{0, 1\}^m \times \prod_{j=1}^n \{0, 1, \dots, h_j\} \times \prod_{i=1}^m \{0, 1, \dots, U_i\}$.

Рассматриваемая система описывается цепью Маркова M с дискретным временем и множеством состояний S . С ненулевой вероятностью за $\sum_{i=1}^m U_i(m + \sum_{j=1}^n h_j)$ переходов из любого состояния достигается состояние $(0, \dots, 0)$, в котором все бункеры пусты, в обработке нет ни одной детали и все машины находятся в состоянии «отказ». Из свойств марковских цепей (см., например, [25], гл. V, § 2) вытекает следующее

Предложение 1. Цепь Маркова M имеет стационарное распределение π , к которому сходится распределение вероятностей состояний линии при $t \rightarrow \infty$ из любого начального состояния.

Средняя производительность линии в стационарном режиме есть $\sum_{i: \alpha_i=(b_j, b_{n+1})} \pi(S_i)/U_i$, где суммирование идет по множеству дуг, входящих в вершину b_{n+1} , а S_i – множество состояний системы, в которых $\alpha_i = 1$ и $y_i = U_i$.

ПРИЛОЖЕНИЕ 2

Опишем упрощенную марковскую модель производственной линии в случае двух последовательных ЕО, предложенную в [6] (аналогичные модели известны из [5, 26]), а также правила агрегирования для пар последовательных и параллельных ЕО, позволяющие использовать эту модель для линий с параллельно-последовательной структурой произвольной сложности.

Аппроксимация модели, описанной в разделе 2 и Приложении 1, состоит в том, что занятый объем бункера и параметр времени предполагаются вещественнозначными, а геометрические распределения вероятностей перехода заменяются экспоненциальными с теми же значениями математических ожиданий. Интенсивность отказов i -й ЕО равна $\lambda_i = 1/T_i^O$, интенсивность восстановления есть $\mu_i = 1/T_i^B$. Скорость обработки деталей на i -й ЕО обозначим через c_i , $c_i = 1/U_i$.

Пусть h – это объем бункера между ЕО. Рассматриваются два типа состояний: множество внутренних состояний $S_{\text{int}} = \{(\alpha_1, \alpha_2, x) : \alpha_1, \alpha_2 \in \{0, 1\}, x \in (0, h)\}$ и множество граничных состояний $S_{\text{fr}} = \{(\alpha_1, \alpha_2, x) : \alpha_1, \alpha_2 \in \{0, 1\}, x \in \{0, h\}\}$, где α_i имеют тот же смысл, что и в Приложении 1. Пусть булевы случайные величины A_i соответствуют значениям α_i , а вещественная случайная величина X – количеству деталей в бункере. Обозначим вероятности состояний из S_{fr} в момент t через $P_{\alpha_1, \alpha_2}(0, t) := \mathbf{P}\{(A_1, A_2, X) = (\alpha_1, \alpha_2, 0) \text{ в момент времени } t\}$ и $P_{\alpha_1, \alpha_2}(h, t) := \mathbf{P}\{(A_1, A_2, X) = (\alpha_1, \alpha_2, h) \text{ в момент времени } t\}$. Обозначим через $F_{\alpha'_1, \alpha'_2}(x, t)$ вероятность того, что в момент времени t случайная величина (A_1, A_2, X) равна (α_1, α_2, x') , где $x' < x$, и предположим, что существуют производные $f_{\alpha_1, \alpha_2}(x, t) = \partial F_{\alpha_1, \alpha_2}(x, t)/\partial x$.

Найденное в [6] стационарное распределение марковского процесса, моделирующего рассматриваемую систему, описывается предельными значениями:

$$P_{\alpha_1, \alpha_2}(0) = \lim_{t \rightarrow \infty} P_{\alpha_1, \alpha_2}(0, t), \quad P_{\alpha_1, \alpha_2}(h) = \lim_{t \rightarrow \infty} P_{\alpha_1, \alpha_2}(h, t), \quad f_{\alpha_1, \alpha_2}(x) = \lim_{t \rightarrow \infty} f_{\alpha_1, \alpha_2}(x, t).$$

При этом внутренние состояния связаны системой уравнений

$$\begin{aligned} 0 &= \lambda_1 f_{10}(x) + \lambda_2 f_{01}(x) - (\mu_1 + \mu_2) f_{00}(x), \\ -c_2 \frac{\partial f_{01}(x)}{\partial x} &= \lambda_1 f_{11}(x) + \mu_2 f_{00}(x) - (\mu_1 + \lambda_2) f_{01}(x), \\ c_1 \frac{\partial f_{10}(x)}{\partial x} &= \lambda_2 f_{11}(x) + \mu_1 f_{00}(x) - (\lambda_1 + \mu_2) f_{10}(x), \\ (c_1 - c_2) \frac{\partial f_{11}(x)}{\partial x} &= \mu_1 f_{01}(x) + \mu_2 f_{10}(x) - (\lambda_1 + \lambda_2) f_{11}(x), \end{aligned}$$

а системы уравнений для граничных состояний зависят от соотношения c_1 и c_2 : $c_1 < c_2$, $c_1 > c_2$ либо $c_1 = c_2 = c$. Для примера опишем случай $c_1 = c_2 = c$:

$$\begin{aligned} P_{10}(0) &= P_{00}(0) = P_{00}(h) = P_{01}(h) = 0, \\ \mu_2 P_{10}(h) &= \lambda_2 P_{11}(h) + cf_{10}(h) = (\lambda_1 + \lambda_2)P_{11}(h), \\ \mu_1 P_{01}(h) &= \lambda_1 P_{11}(0) + cf_{01}(0) = (\lambda_1 + \lambda_2)P_{11}(0), \\ cf_{01}(h) &= \lambda_1 P_{11}(h), \\ cf_{10}(0) &= \lambda_2 P_{11}(0). \end{aligned}$$

Решение этой системы уравнений дает выражения для $f_{\alpha_1, \alpha_2}(x)$, $P_{\alpha_1, \alpha_2}(0)$ и $P_{\alpha_1, \alpha_2}(h)$.

Правило R_1 . На основе описанной модели и полученных вероятностей ее состояний в стационарном режиме находится «эквивалентная» ЕО, которая имеет близкие характеристики в стационарном режиме и может заменить две ЕО и бункер между ними. Для этого введем состояние отказа для системы из двух ЕО и бункера между ними. Определим это состояние со стороны менее производительной ЕО, так как она простаивает меньше и погрешность расчета будет меньше. Под отказом будем понимать состояние, когда менее производительная ЕО не находится в состоянии «работа».

ЕО, заменяющая две последовательные ЕО, имеет параметры λ' , μ' и c' . Интенсивности λ' и μ' вычисляются на основе интенсивностей перехода между состояниями «отказ» и «работа» для рассматриваемой двухмашинной подсистемы с учетом стационарного распределения вероятностей для двух последовательных машин и бункера между ними [4]. Например, при $c_2 > c_1$ имеем

$$\begin{aligned} c' &= c_1, \quad \lambda' := \lambda_1 + \frac{P_{10}(h)\mu_2}{F_{11}(h) + F_{10}(h) + P_{11}(0)}, \\ \mu' &:= \mu_1 + \frac{P_{10}(h)(\mu_2 - \mu_1)}{P_{10}(h) + P_{01}(0) + F_{01}(h) + F_{00}(h)}. \end{aligned}$$

Среднее число деталей в бункере в стационарном режиме имеет вид [6]

$$q = \sum_{\alpha_1, \alpha_2 \in \{0,1\}} \left(\int_0^h x f_{\alpha_1 \alpha_2}(x) dx + h P_{\alpha_1 \alpha_2}(h) \right).$$

Правило R_2 . Параметры ЕО, заменяющей две параллельные ЕО [18]:

$$\begin{aligned} \lambda' &:= \lambda_1 \frac{\mu_2}{\mu_2 + \lambda_2} + \lambda_2 \frac{\mu_1}{\mu_1 + \lambda_1}, \quad c' := c_1 + c_2, \\ \mu' &:= \frac{\lambda^*}{(c_1 + c_2)/v' - 1}, \quad \text{где } v' = \frac{c_1}{1 + \lambda_1/\mu_1} + \frac{c_2}{1 + \lambda_2/\mu_2}. \end{aligned}$$

ПРИЛОЖЕНИЕ 3

Приведем описание линий, построенных на основе примеров из [18]. В задачах AS1, AS2 и AS6 линии имеют последовательную структуру, их параметры приведены в табл. 2,3.

В задачах AS7, AS8 линии состоят из 10 ЕО и имеют последовательно-параллельную структуру. Их схемы представлены на рис. 4. Параметры линий для задач AS7, AS8 представлены в табл. 4.

Таблица 2. Параметры задачи AS1

Бункеры		ЕО			
i	d_i	j	T_j^O	T_j^B	U_i
1	20	1	244,2	150	10
2	17	2	255,3	300	10
3	38	3	176	75	10
4	48	4	184	600	10
		5	192	450	10

Таблица 3. Параметры задач AS2 и AS6

AS2						AS6					
Бункеры		ЕО				Бункеры		ЕО			
i	d_i	j	T_j^O	T_j^B	U_i	i	d_i	j	T_j^O	T_j^B	U_i
1	0	1	10000	440	22	1	60	1	29880	22000	385
2	50	2	20000	440	23	2	60	2	29880	22000	426
3	20	3	5000	430	22	3	50	3	876000	22300	330
4	50	4	40000	520	23	4	70	4	29880	22000	372
5	0	5	30000	430	24	5	60	5	33250	27500	316
6	80	6	2442	440	22	6	80	6	144000	8500	340
7	20	7	1840	520	23	7	45	7	102300	74000	340
8	100	8	1680	430	21	8	25	8	113300	7200	340
9	100	9	2208	920	24	9	35	9	540000	60000	380
		10	2464	780	22	10	80	10	538800	349000	350
						11	40	11	5064000	73700	400
						12	45	12	468000	306000	400
						13	65	13	1032000	54000	319
								14	45600	31120	319

СПИСОК ЛИТЕРАТУРЫ

1. *Tempelmeier H.* Practical considerations in the optimization of flow production systems // Int. J. Product. Res. 2003. V. 41. № 1. P. 149-170.
2. *Patchong A., Lemoine T., Kern G.* Improving car body production at PSA Peugeot Citroen // Interfaces. 2003. V. 33 № 1. P. 36-49.
3. *Dallery Y., Gershwin S.B.* Manufacturing flow line systems: a review of models and analytical results // Queueing Syst. 1992. V. 12. № 1-2. P. 3-94.
4. *Долгий А.Б., Свирич Ю.П.* Модели оценки вероятностной производительности автоматизированных технологических комплексов // Вест. АН Беларуси. 1995. № 1. С. 59-67.

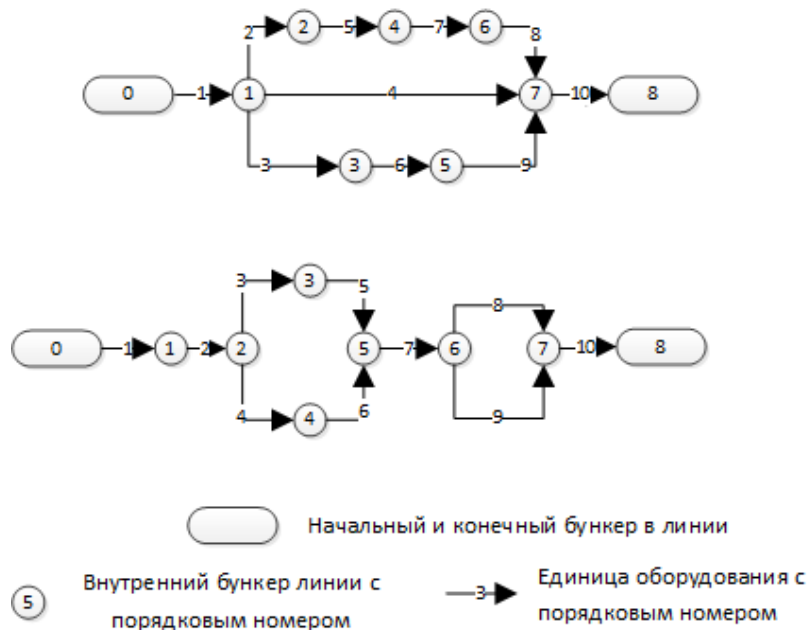


Рис. 4. Графы линий в примерах AS7 (сверху) и AS8 (снизу).

5. Левин А.А., Пасько Н.И. Расчет производительности автоматических линий // Станки и инструмент. 1969. № 8. С. 8-10.
6. Dubois D., Forestier J.-P. Productivité et en cours moyen d'un ensemble de deux machines séparées par une zone de stockage // RAIRO Automat. 1982. V. 16. № 2. P. 105-132.
7. Li J., Meerkov S.M. Production Systems Engineering. N.Y.: Springer, 2009.
8. Altiparmak A., Bugak A., Dengiz B. Optimization of buffer sizes in assembly systems using intelligent techniques // Proc. 2002 Winter Simulat. Conf. 2002. P. 1157-1162.
9. D'Souza K., Khator S. System reconfiguration to avoid deadlocks in automated manufacturing systems // Comput. Indust. Engin. 1997. V. 32. P. 445 - 465.
10. Hamada M., Martz H., Berg E., Koehler A. Optimizing the product-based availability of a buffered industrial process// Reliabilit. Engin. Syst. Safety. 2006. V. 91. P. 1039 - 1048.
11. Abdul-Kader W. Capacity improvement of an unreliable production line - an analytical approach// Comput. Oper. Res. 2006. V. 33. P. 1695 - 1712.
12. Dolgui A., Ereemeev A., Kolokolov A., Sigaev V. A genetic algorithm for the allocation of buffer storage capacities in a production line with unreliable machines// J. Math. Modeling Algorithms. 2002. V. 1. P.89-104.
13. Chehade H., Yalaoui F., Amodeo L., De Guglielmo P. Optimisation multiobjectif pour le problème de dimensionnement de buffers // J. Decision Syst. 2009. V. 18. P. 257-287.

Таблица 4. Параметры задач AS7 и AS8

AS7						AS8					
Бункеры		ЕО				Бункеры		ЕО			
i	d_i	j	T_j^O	T_j^B	U_i	i	d_i	j	T_j^O	T_j^B	U_i
1	15	1	50000	12000	1000	1	1300	1	87000	27000	23
2	10	2	48000	2000	3450	2	200	2	77000	22000	27
3	15	3	55000	9000	2780	3	0	3	580000	18000	38
4	10	4	39000	6000	3030	4	0	4	410000	12500	30
5	25	5	75000	10000	3333	5	0	5	580000	18000	38
6	10	6	59000	11000	2560	6	2000	6	410000	12500	30
7	10	7	28000	8000	3030	7	0	7	725000	21000	20
		8	35000	8000	3125			8	550000	14000	40
		9	65000	35000	2174			9	430000	24000	43
		10	20000	4000	800			10	270000	22000	33

14. *Zitzler E., Laumanns M., Thiele L.* SPEA2: Improving the Strength Pareto Evolutionary Algorithm // Technic. Report 103, Comput. Engin. Commun. Networks Lab, Swiss Federal Institute Technol., Zurich, 2001.
15. *Deb K., Pratap A., Agarwal S., Meyarivan T.* A fast and elitist multiobjective genetic algorithm: NSGA-II // IEEE Transact. Evolut. Comput. 2002. V. 6. № 2. P. 182–197.
16. *Cruz F.R.B., Van Woensel T., Smith J.M.* Buffer and throughput trade-offs in M/G/1/K queuing networks: A bicriteria approach // Int. J. Product. Econom. 2010. V. 125. P. 224–234.
17. *Севастьянов Б.А.* Задача о влиянии емкости бункеров на среднее время простоя автоматической линии станков // Теория вероятностей и ее применения. 1962. Т 7. № 4. С. 438–447.
18. *Ancelin B., Semery A.* Calcul de la productivité d'une ligne intégrée de fabrication // RAIRO Autom., Productiq. Inform. Industrielle. 1987. V. 21. P. 209-238.
19. *Terracol C., David R.* Performance d'une ligne composée de machines et de stocks intermédiaires // RAIRO Automatiq., Productiq. Informatiq. Industrielle. 1987. V. 21. P. 239-262.
20. *Dolgui A., Ereemeev A., Kovalyov M.Y., Sigaev V.* Complexity of buffer capacity allocation problems for production lines with unreliable machines // J. Math. Modell. Algorithms. 2013. V. 12. P.155-165.
21. *Brockhoff D., Friedrich T., Neumann F.* Analyzing hypervolume indicator based algorithms // Proc. Parallel Probl. Solving from Nature - PPSN X: 10th Int. Conf. 2008. V. 5199. Berlin: Springer, 2008. 651-660

22. *Laumanns M., Thiele L., Zitzler E., Welzl E., Deb K.* Running time analysis of a multi-objective evolutionary algorithm on a simple discrete optimization problem // *Parallel Probl. Solving From Nature*. 2002. V. 2439. Berlin: Springer, 2002. 44-53
23. *Белоус В.В., Грошев С.В., Карпенко А.П., Шибитов И.А.* Программные системы для оценки качества Парето-аппроксимации в задаче многокритериальной оптимизации. Обзор // *Наука и образование: ФГБОУ ВПО "МГТУ им. Н.Э. Баумана"*. 2014. Эл № ФС 77 - 48211. С.300-320.
24. *Zitzler E., Brockhoff D., Thiele L.* The hypervolume indicator revisited: On the design of Pareto-compliant indicators via weighted integration // *Proc. Conf. Evolut. Multi-Criter. Optim. (EMO 2007)*, LNCS, V. 4403, Berlin: Springer. 2007. P. 862–876.
25. *Дуб Дж.Л.* Вероятностные процессы. М.: ИИЛ, 1956.
26. *Gershwin S.B., Schick I.C.* Continuous model of an unreliable two-stage material flow system with a finite interstage buffer // *Report LIDS-R-1039*, Massachusetts Institute Technol., Cambridge, 1980.