

# Эволюционный подход к моделированию распределительных процессов

Н.Н. Венцов

## Актуальность темы

Существенный класс научно-практических задач связан с распределением потоков ресурсов. Распределяемые ресурсы могут иметь различную структуру и характеристики, обусловленные природой их происхождения. В этой связи необходимо разрабатывать подходы, обеспечивающие эффективное распределение ресурсов не зависимо от размерностей решаемых задач, а так же особенностей моделируемой в процессе распределения предметной области [1-6].

**Постановка задачи.** Имеется поток ресурсов  $S$ , подлежащий распределению между  $N$  производствами (рис.1). Потери ресурса при его доставке к  $j$ -му производству, определяются величиной  $p_j$ ,  $j \in \{1, 2, 3, \dots, N\}$ . Технологический минимум ресурсов требуемый  $j$ -му производству определяется величиной  $\xi_j$ , а технологический максимум –  $a_j$ . Прибыльность  $j$ -го производства зависит от объема доступных ресурсов  $x_j$  и описывается нелинейной функцией  $f_j(x_j)$ .

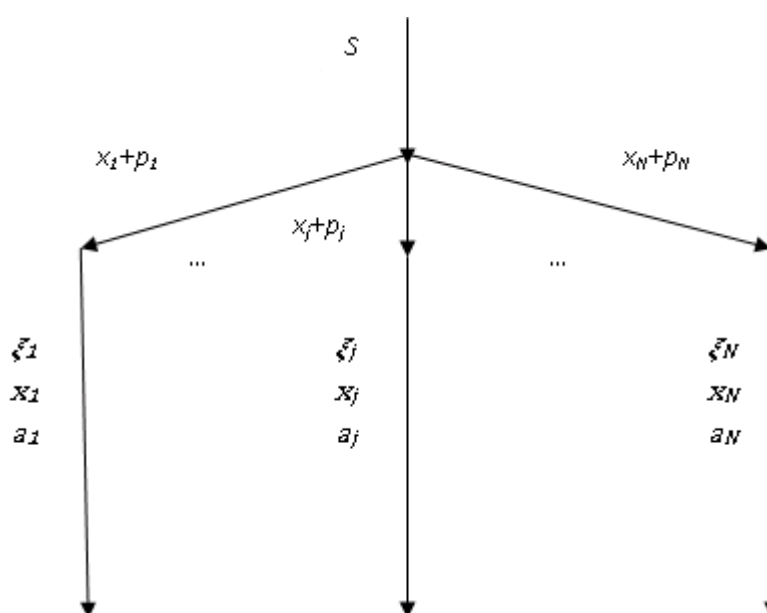


Рис. 1 Схема процесса распределения ресурсов

Необходимо распределить ресурсы  $S$  таким образом чтобы максимизировать получаемую прибыль которая определяется по формуле:

$$\sum_{j=1}^N f_j(x_j) \rightarrow \max \quad (1)$$

Ограничения, накладываемые на условия и решения задачи, описываются выражениями:

$$\sum_{j=1}^N (x_j + p_j) \leq S, \quad (2)$$

$$\sum_{j=1}^N a_j < S, \quad (3)$$

$$\sum_{j=1}^N \xi_j < S, \quad (4)$$

$$\xi_j < x_j < a_j, \quad j \in \{1, 2, 3, \dots, N\}, \quad (5)$$

$$\xi_j > 0, \quad j \in \{1, 2, 3, \dots, N\}. \quad (6)$$

Из неравенства (5) следует что на любое производство должно быть подано некоторое минимально допустимое количество ресурса  $\xi_j + p_j$ : связанное как с наличием технологической нижней границы перерабатываемых объемов в каждом отдельном производстве  $\xi_j$ , так и с потерями доставки ресурса  $p_j$ . Таким образом минимальный суммарный объем запаса ресурсов определяется как:

$$P = \sum_{j=1}^N (\xi_j + p_j). \quad (7)$$

По этой причине объем фактически распределяемого ресурса  $\hat{S}$  определяется по формуле:

$$\hat{S} = S - P. \quad (8)$$

На основании выражений (1)-(5) можно говорить о том, что решение задачи заключается в нахождении точки оптимума в  $N$ -мерном пространстве решений. На практике функционал (1) может не обладать свойствами, облегчающими его оптимизацию такими как линейность, непрерывность, дифференцируемость в любой точке и т.д. По этой причине использование классических методов оптимизации не целесообразно.

**Используемый подход.** Исследуем возможность применения метода «оптимизации с использованием роя частиц» (*Particle Swarm Optimization, PSO*) [7-10] для решения поставленной задачи. *PSO*-метод базируется на понятии популяции, и моделирует поведение птиц в стае и косяков рыб. Стратегия поведения особей (частиц) в популяции (рое) состоит в стремлении превзойти достижения соседних частиц и улучшить собственные. Перемещения частиц внутри области поиска обуславливаются

их природным стремлением конкурировать между собой. Выбор траектории движения осуществляются частицей на основе личного опыта, а также опыта её соседей.

Как утверждалось ранее, решение поставленной задачи состоит в нахождении точки оптимума в  $N$ -мерном пространстве решений. Если предположить что частица движется в  $N$ -мерном пространстве, то ее координаты можно описать вектором (кортежем)  $X = \langle x_1, x_2, \dots, x_j, \dots, x_N \rangle$ . Значение каждого элемента кортежа определяет количество ресурсов выделенных соответствующему производству, например величина  $x_1$  определяет количество ресурсов выделенных первому процессу,  $x_2$  - второму и т.д. С помощью кортежа  $X_i(t) = \langle x_{i,1}(t), x_{i,2}(t), \dots, x_{i,j}(t), \dots, x_{i,N}(t) \rangle$  обозначим позицию  $i$ -ой частицы в пространстве поиска решений в момент времени  $t$ . Для простоты изложения, факт вычисления функционала (1) на основе кортежа  $X_i(t) = \langle x_{i,1}(t), x_{i,2}(t), \dots, x_{i,j}(t), \dots, x_{i,N}(t) \rangle$  обозначим как:

$$F[X_i(t)] \rightarrow \max \quad (9)$$

Процесс поиска решений данным методом начинается с генерации частиц. Начальное состояние частицы  $i$ , в нулевой момент времени, описывается кортежем  $X_i(0) = \langle x_{i,1}(0), x_{i,2}(0), \dots, x_{i,j}(0), \dots, x_{i,N}(0) \rangle$  и определяется следующим образом  $X_i(0) = U(\xi, A)$ . Функция  $U(\xi, A)$  для каждого измерения генерирует случайное число из диапазона  $[\xi_j, a_j]$ .

Позиция  $i$ -ой частицы в пространстве поиска решений изменяется добавлением скорости  $V_i(t) = \langle v_{i,1}(t), v_{i,2}(t), \dots, v_{i,j}(t), \dots, v_{i,N}(t) \rangle$  к текущей позиции:

$$X_i(t + 1) = X_i(t) + V_i(t + 1) \quad (10)$$

В разновидности *gbest* PSO-метода каждая частица тяготеет к лучшему решению целого роя поэтому скорость  $i$ -ой частицы в  $j$ -ом измерении определяется по формуле:

$$v_{i,j}(t + 1) = v_{i,j}(t) + c_1 r_1(t) [y_{i,j}(t) - x_{i,j}(t)] + c_2 r_2(t) [y_j^*(t) - x_{i,j}(t)] \quad (11)$$

где:  $v_{i,j}(t)$ - скорость  $i$ -ой частицы в  $j$ -ом измерении в момент времени  $t$ ;  
 $x_{i,j}(t)$  - координаты частицы  $i$  в измерении  $j$ ;  $c_1$  и  $c_2$  – положительные константы ускорения, варьирующие когнитивную и социальную компоненты скорости частицы;  $r_1$  и  $r_2$  - случайные переменные, принимающие значения 0 или 1;  $y_{i,j}(t)$ - координата наилучшей достигнутой позиции частицы  $i$  в  $j$ -ом измерении;  $y_j^*(t)$ - координата наилучшей достигнутой позиции роя в  $j$ -ом измерении.

В соответствии с формулами (10) и (11) в случае если:

- $x_{i,j}(t + 1) < \xi_j$  то  $x_{i,j}(t + 1) := \xi_j$ ;
- $x_{i,j}(t + 1) > a_j$  то  $x_{i,j}(t + 1) := a_j$ .

Кортеж  $Y_i = \langle x_{i,1}, x_{i,2}, \dots, x_{i,j}, \dots, x_{i,N} \rangle$  отображает наилучшую позицию частицы  $i$ , которую она посещала, начиная с первой итерации. Обозначим функцию  $\sum_{j=1}^N f_j(x_j)$  из выражения (1) через  $F(X)$ , тогда в момент времени  $t+1$  следующая оптимальная позиция частицы  $i$  рассчитывается по формуле:

$$\begin{cases} Y_i = Y_i & \text{если } F(X(t + 1)) \leq F(Y) \\ Y_i = X(t + 1) & \text{если } F(X(t + 1)) > F(Y) \end{cases} \quad (12)$$

Величина  $\max(Y_i) \quad i \in \{1, 2, 3, \dots, M\}$ , где  $M$  число особей в рое, определяет наилучшее решение полученное роем.

**Предлагаемый алгоритм.** На основе изложенного выше метода был разработан эвристический алгоритм оптимизации распределительных процессов в переработке ресурсов.

Первый такт работы алгоритма начинается с процесса генерации особей, образующих пчелиный рой. Координаты генерируемых особей соответствуют ограничениям (2)-(6). Каждой особи, на основе ее координат, ставится в соответствие значение целевой функции (1). Скорости всех сгенерированных особей равны нулю.

Следующая группа блоков (команд) выполняется итеративно:

- блок «Определение статистики» описывает процесс нахождения величины  $y_j^*(t)$ - координаты наилучшей достигнутой позиции роя найденной

за  $t$  итераций и величины  $y_i(t)$ - координаты наилучшей достигнутой позиции частицы найденной за  $t$  итераций. Под наилучшей позицией понимается такое положение частицы в пространстве, при котором значение функционала (1) максимально. Таким образом величина  $y^*(t)$  определяется для всего роя, а величина  $y_i(t)$  – для каждой частицы в отдельности;

- блок «Расчет скоростей» описывает процесс расчета для каждой особи роя по формуле (11) новой скорости в каждом из пространств;

- блок «Перемещение особей» описывает процесс расчета для каждой особи роя по формуле (10) новых координат;

- блок «Расчет ЦФ» соответствует процессу вычисления целевой функции (1) для каждой особи роя, на основе новых координат.

Критерием останова алгоритма является выполнение заданного числа итераций.

**Экспериментальная часть.** Исследуем рост погрешности алгоритма на размерностях 1-10, приравняв число итераций алгоритма и число особей роя 5. Часть результатов исследования приведена в таблице 1.

Таблица № 1

Погрешности алгоритма на задачах размерности 1-10

№ п/п	Размерность	Число особей	Число итераций	Погрешность
1	1	5	3	0,033
2	2	5	5	0,0435
3	3	5	5	0,087
4	4	5	5	0,0675
5	5	5	5	0,072
6	6	5	5	0,064
7	7	5	5	0,07374
8	8	5	5	0,06234
9	9	5	5	0,07267
10	10	5	5	0,095

Погрешность алгоритма определялась по формуле:

$$\gamma = \frac{F(Y_{opt}) - F(Y)}{F(Y_{opt})} \text{ где,} \quad (13)$$

-  $Y_{opt}$  - оптимальное решение задачи,  $Y$  – наилучшее из предложенных алгоритмом решение задачи.

Из таблицы 1 следует что в целом лавинообразное накопление погрешности отсутствует. Данное утверждение позволяет сделать предположение о том, что рост размерностей решаемых задач не требует пропорционального увеличения вычислительных ресурсов требуемых для получения приемлемого решения.

Как следует из условия, каждое решение задачи соответствует некоторому распределению ресурсов между производствами. В процессе работы алгоритма решения итеративно модифицируются, соответственно изменяя распределение ресурсов. Рассмотрим динамику изменения распределений ресурсов между производствами, в зависимости от числа итераций и количества особей роя. Результаты экспериментов частично отражены в таблице 2.

Таблица № 2

Распределение ресурсов

N п/п	Число особей	Число итераций	x1	x2	x3	x4	Отклонение
1	2	2	0,2	0,2	2,2	0,2	0,183125
2	3	3	1,2	1,2	0,2	1,2	0,096875
3	4	4	1,2	1,2	0,2	1,2	0,096875
4	5	5	2,2	1,2	2,2	0,2	0,074375
5	6	6	2,2	2,2	2,2	0,2	0,063125
6	7	7	2,2	2,2	2,2	0,2	0,063125
7	8	8	2,2	2,2	2,2	0,2	0,063125
8	9	9	2,2	0,32	2,2	0,8	0,055625
9	10	10	2,2	0,39	2,2	1,94	0,051875
10	11	11	2,2	0,53	2,2	1,91	0,045625
11	12	12	1,66	0,88	2,2	1,16	0,043125
12	13	13	2,2	2,04	1,42	0,59	0,048125
13	14	14	2,2	1,2	2,2	1,2	0,025625
14	15	15	2,2	1,2	2,2	1,2	0,025625
15	16	16	2,2	0,53	2,19	0,81	0,02375
16	17	17	1,99	1,66	2,2	1,64	0,006875
17	18	18	1,97	1,74	2,2	1,6	0,006875
18	19	19	1,46	1,52	1,8	1,81	0,0125
19	20	20	2,2	2,17	1,58	1,89	0,00375

Как следует из таблицы 2 ресурсы распределялись между 4 процессами ( $x_1 - x_4$ ). Априорно было установлено что оптимальному распределению ресурсов (решению задачи) соответствует кортеж  $\langle 2,2,2,2 \rangle$ . Графически изменение распределения ресурсов между процессами приведено на рис. 2.

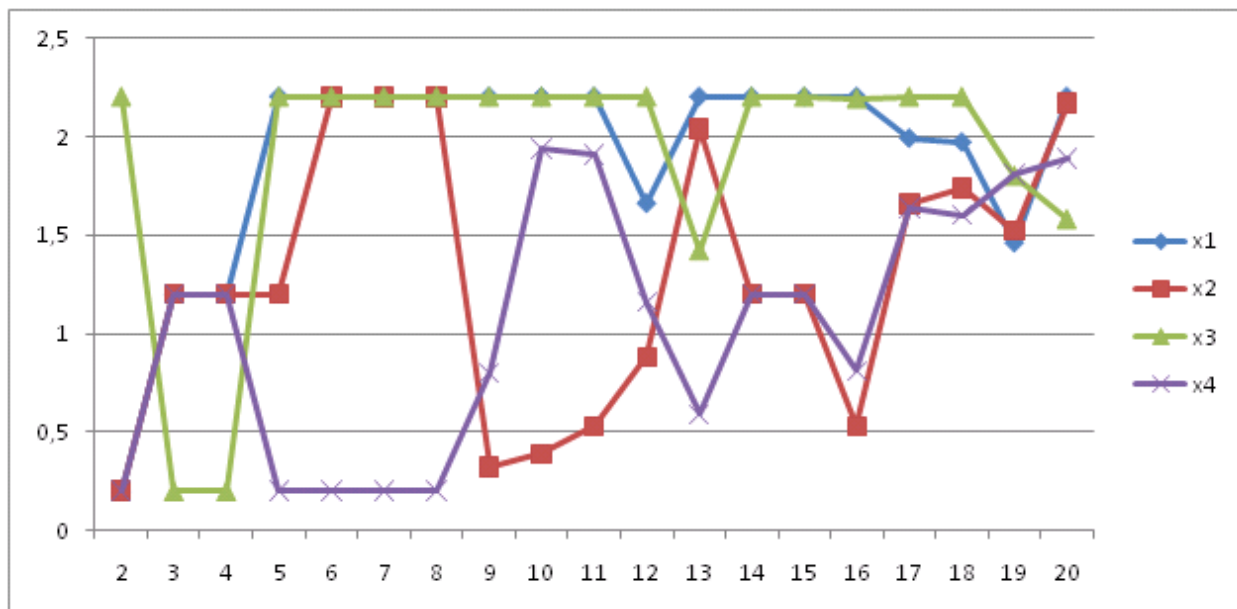


Рис.2 Зависимость распределения ресурсов от числа итераций и мощности роя

На основании данных представленных в таблице 2 и рис.2 можно предположить что с увеличением числа итераций и количества особей в рое получаемое распределение ресурсов стремится к оптимальному т.е. погрешность алгоритма снижается.

В рассматриваемом ниже примере поток ресурсов распределяется между четырьмя процессами. График временной сложности алгоритма (ВСА), на задачах размерности 1-20, приведен на рис.3.

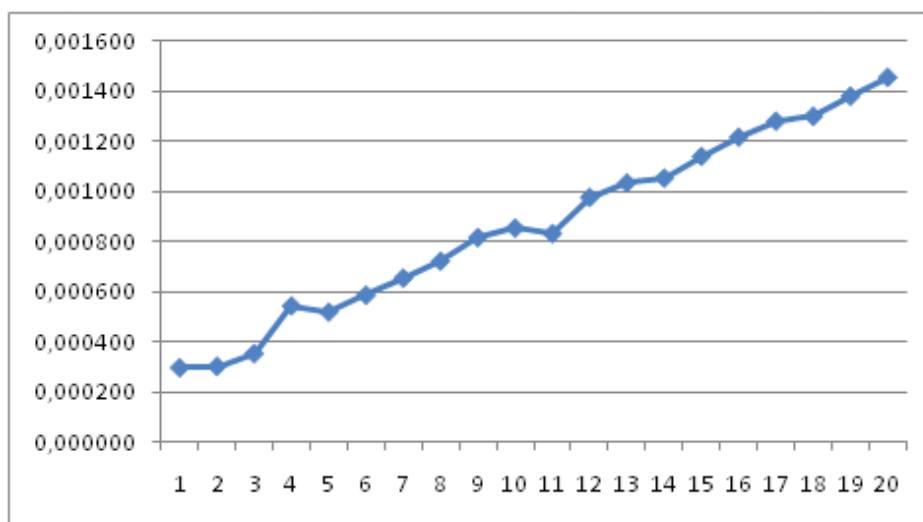


Рис. 3 График ВСА

На основании рис. 3 можно предположить что для задач размерности 1-20 ВСА возрастает линейно.

**Выводы.** На основе *PSO*- метода разработан эвристический алгоритм оптимизации распределительных процессов, позволяющий решать многомерные оптимизационные задачи. Предлагаемый алгоритм способен выходить из областей субоптимальных решений но в тоже время необходимо разработать адаптивные механизмы, препятствующие преждевременной сходимости. На задачах размерности 1-10 алгоритм позволяет определять субоптимальные решения погрешность которых не превышает 0,1

#### Литература:

1. Золотарев А.А., Дидковский Д.О. Оптимальная параметризация в задачах распределения ресурсов [Текст]// Вестник Донского государственного технического университета, Т.9. ч.2 – Ростов-на-Дону: Изд. центр ДГТУ, 2009. с.5-12.

2. Лебедев Б.К., Лебедев В.Б. Поисковые процедуры канальной трассировки, базирующиеся на моделировании адаптивного поведения роя частиц в пространстве решений с неупорядоченным лингвистическим шкалированием [Текст]//Известия ЮФУ. Технические науки. -2009. -№ 12 (101). -С. 15-22.



3. Курейчик В.М. Биоинспирированный поиск с использованием сценарного подхода [Текст]//Известия ЮФУ. Технические науки. -2010. -№ 7 (108). -С. 7-12.

4. Чернышев Ю.О., Венцов Н.Н., Мухтаров С.А. Алгоритм статической оптимизации передачи данных [Текст]//Известия ЮФУ. Технические науки. - 2013. -№ 7 (144). -С. 136-141.

5. Побегайлов О.А., Шемчук А.В. Современные информационные системы планирования в строительстве [Электронный ресурс] // Инженерный вестник Дона, 2012. - №2 <http://ivdon.ru/magazine/archive/n3y2012/963> (доступ свободный) – Загл. с экрана. – Яз. рус.

6. Сироткин А.В. Приоритетное планирование процессов информационного обеспечения в АСУП [Электронный ресурс]// Инженерный вестник Дона, 2012. - №1. <http://www.ivdon.ru/magazine/archive/n1y2012/629> (доступ свободный) – Загл. с экрана. – Яз. рус.

7. Eberhart R., Kennedy J. A New Optimizer using Particle Swarm Theory // In Proceedings of the Sixth International Symposium on Micro machine and Human Science 1995 – P. 39-43

8. Engelbrecht A. Computational intelligence: an introduction – John Wiley and Sons Ltd., 2007 – 597p.

9. Abraham A., Grosan G. Swarm Intelligence in Data Mining –Springer, 2006 – 267p.

10. Чернышев Ю.О., Венцов Н.Н., Мухтаров С.А. Алгоритм статической оптимизации передачи данных [Текст] //Известия ЮФУ. Технические науки. – 2013. – № 7. – С. 136-141.