

Метод имитация отжига

Алгоритм имитации отжига

Метод отжига служит для поиска глобального минимума некоторой функции $f(x)$, заданной для x некоторого пространства S , дискретного или непрерывного. Элементы множества S представляют собой состояние воображаемой физической системы («энергетические уровни»), а значения функции f в этих точках используется как энергия системы $E = f(x)$. В каждый момент предполагается заданная температура системы T , как правило, уменьшающаяся с течением времени. После попадания в состояние x при температуре T , следующее состояние системы выбирается в соответствии с заданным порождающим семейством вероятностных распределений $\mathcal{G}(x, T)$, которое при фиксированных x и T задает случайный элемент $G(x, T)$ со значениями в пространстве S . После генерации нового состояния $x' = G(x, T)$, система с вероятностью $h(\Delta E, T)$ переходит к следующему состоянию x' , в противном случае процесс генерации x' повторяется. Здесь ΔE обозначает приращение функции $f(x') - f(x)$. Величина $h(\Delta E, T)$ называется вероятностью принятия нового состояния.

Как правило, в качестве функции $h(\Delta E, T)$ выбирается либо точное значение соответствующей физической величины

$$h(\Delta E, T) = \frac{1}{1 + \exp(\Delta E/T)}$$

либо приближенное значение

$$h(\Delta E, T) = \exp(-\Delta E/T)$$

Вторая формула используется наиболее часто. При ее использовании $h(\Delta E, T)$ оказывается больше единицы в случае $\Delta E < 0$, и тогда соответствующая вероятность считается равной 1. Таким образом, если новое состояние дает лучшее значение оптимизируемой функции, то переход в это состояние перейдет в любом случае.

$$G(x, T) = \begin{cases} 1, & f(x') - f(x) < 0 \\ \exp(-\Delta E/T) & f(x') - f(x) \geq 0 \end{cases}$$

Конкретная схема метода отжига задается следующими параметрами:

- Выбором закона изменения температуры $T(k)$, где k – номер шага.
- Выбором порождающего семейства распределений $G(x, T)$.
- *Выбором функции вероятности принятия $h(\Delta E, T)$.*

Алгоритм:

1) Случайным образом выбирается начальная точка $x = x_0$, $x_0 \in S$. Текущее значение энергии E устанавливается в значение $f(x_0)$.

Для большинства задач начальное решение будет случайным. На самом первом шаге оно помещается в текущее решение. Другая возможность заключается в том, чтобы загрузить в качестве начального решения уже существующее, возможно, то самое, которое было найдено во время предыдущего поиска. Это предоставляет алгоритму базу, на основании которой выполняется поиск оптимального решения задачи.

2) k -я итерация основного цикла состоит из следующих шагов:

а) Сравнить энергию системы E в состоянии x с найденным на текущий момент глобальным минимумом. Если $E = f(x)$ меньше, то изменить значение глобального минимума.

Оценка решения состоит из декодировки текущего решения и выполнения нужного действия, позволяющего понять его целесообразность для решения данной задачи. Обратите внимание, что закодированное решение может просто состоять из набора переменных. Они будут декодированы из существующего решения, а затем эффективность решения будет оценена на основании того, насколько успешно удалось решить данную задачу.

а) Сгенерировать новую точку $x' = G(x, T(k))$.

Случайный поиск решения начинается с копирования текущего решения в рабочее. Затем мы произвольно модифицируем рабочее решение. Как именно модифицируется рабочее решение, зависит от того, каким образом оно представляется (кодируется). Представьте себе кодировку задачи коммивояжера, в которой каждый элемент представляет собой город. Чтобы выполнить поиск по рабочему решению, мы берем два элемента и переставляем их. Это позволяет сохранить целостность решения, так как при этом не происходит повторения или пропуска города. После выполнения поиска рабочего решения мы оцениваем решение, как было описано ранее. Поиск нового решения основан на методе Монте-Карло (то есть случайным образом).

б) Вычислить значение функции в ней $E' = f(x')$.

с) Сгенерировать случайное число α из интервала $[0; 1]$.

д) Если $\alpha < h(E' - E, T(k))$, то установить $x \leftarrow x'$, $E \leftarrow E'$ (обратная импликация) и перейти следующей итерации. Иначе повторить шаг б), пока не будет найдена подходящая точка x' .

Однако если рабочее решение хуже, чем текущее решение, мы определяем критерий допуска, чтобы выяснить, что следует сделать с текущим рабочим решением. Вероятность допуска основывается на уравнении $h(\Delta E, T) = \exp(-\Delta E/T)$ (которое, в свою очередь, базируется на законе термодинамики).

После ряда итераций по алгоритму при данной температуре мы ненамного снижаем ее. Существует множество вариантов снижения температуры. Но часто используется простая геометрическая функция:

$$T(k + 1) = \alpha T(k)$$

Константа α меньше единицы. Возможны и другие стратегии снижения температуры, включая линейные и нелинейные функции.

Известны следующие модификации этого алгоритма:

Модификация А. На шаге 2е переход к следующей итерации происходит и в том случае, если точка x' не являлась подходящей. При этом следующая итерация начинается с точки x , но уже с новым значением температуры.

Модификация Б. В качестве оценки точности минимума возвращается последнее значение x . Это может незначительно ускорить алгоритм в случае большой размерности S , но с небольшой вероятностью может привести к тому, что будет получено худшее решение (особенно, если температура к моменту завершения остается значительно больше нуля).

Модификация В. На шаге 2b x' вычисляется рекуррентно с использованием формулы $x' = G(x', T(k))$. Изначально на шаге 1 устанавливается $x' \leftarrow x_0$. Это позволяет избежать «застревания» алгоритма, однако такая реализация теряет множество преимуществ метода отжига, т.к. не очень сильно отличается от обычного случайного поиска.

Пример. Найти длину гамильтонова цикла S_4 в полном графе K_6 (рис.1) после четырех циклов решения задачи методом отжига.

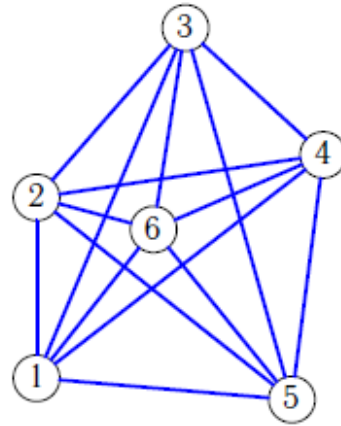


Рис. 1. Полный граф

Дана начальная последовательность прохождения маршрута $V = [1, 4, 3, 5, 6, 2, 1]$, последовательность замен вершин $Z = [V_4 \rightleftharpoons V_6], [V_5 \rightleftharpoons V_6], [V_3 \rightleftharpoons V_5], [V_6 \rightleftharpoons V_2]$, где V_i – номер элемента в V и выпавшие в процессе счета случайные числа $P = 64, 62, 83, 71$. Снижение температуры происходит по закону $T_{k+1} = 0,5T_k$ от $T_1 = 100$. Верхнюю границу вероятности перехода на худшее решение рассчитывать по формуле $h(\Delta E, T) = \exp(-\Delta E/T)$.

Зададим длину ребра.

Ребро	$L_{i,j}$	Ребро	$L_{i,j}$
1 – 2	19	2 – 6	13
1 – 3	41	3 – 4	20
1 – 4	39	3 – 5	41
1 – 5	27	3 – 6	22
1 – 6	20	4 – 5	26
2 – 3	24	4 – 6	20
2 – 4	31	5 – 6	23
2 – 5	35		

Решение.

1. Находим длину первоначального маршрута $V_0 = [1, 4, 3, 5, 6, 2, 1]$ (рис.2.)

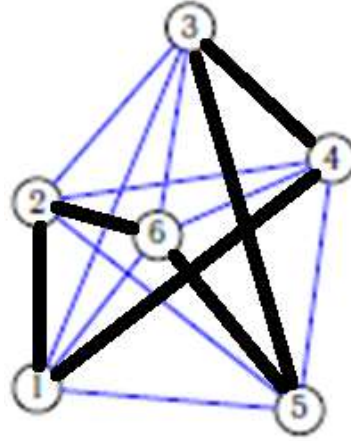


Рис. 2. Первоначальный маршрут V_0

$$S_0 = L_{1,4} + L_{4,3} + L_{3,5} + L_{5,6} + L_{6,2} + L_{2,1} = 39 + 20 + 41 + 23 + 13 + 19 = 155$$

Меняем порядок прохождения вершин. По условию задачи меняем местами четвертый и шестой элемент списка прохождения, т.е. вершины 5 и 2. Получаем новый список $V_1 = [1, 4, 3, 2, 6, 5, 1]$ (рис. 3).

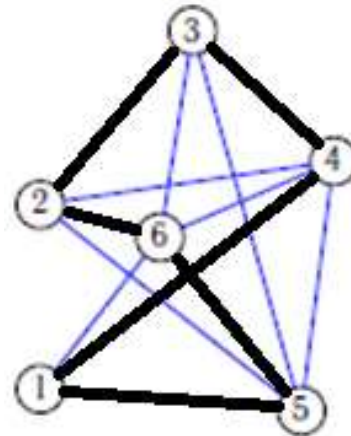


Рис. 3. Маршрут V_1

Находим длину соответствующего маршрута

$$S_1 = L_{1,4} + L_{4,3} + L_{3,2} + L_{2,6} + L_{6,5} + L_{5,1} = 39 + 20 + 24 + 13 + 23 + 27 = 146$$

Вычисляем приращение длины $\Delta S_1 = S_1 - S_0 = -9$. Так как приращение отрицательное, т.е. длина уменьшилась (решение улучшилось), то маршрут V_1 принимается. Одновременно уменьшается температура $T_2 = 0,5T_1 = 50$.

1. Выполняем очередную замену. Меняем местами пятый и шестой элемент списка порождения вершин. Получаем последовательность $V_2 = [1, 4, 3, 2, 5, 6, 1]$ (рис. 4).

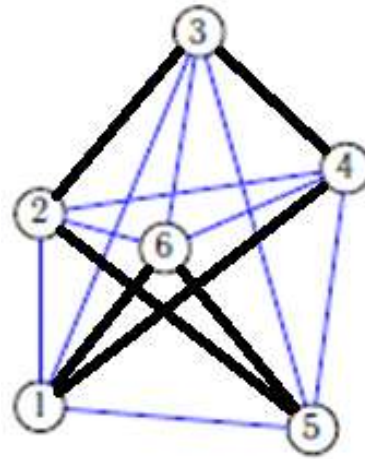


Рис. 4. Маршрут V_2

Находим длину маршрута.

$$S_2 = L_{1,4} + L_{4,3} + L_{3,2} + L_{2,5} + L_{5,6} + L_{6,1} = 39 + 20 + 24 + 35 + 23 + 20 = 161$$

Вычисляем приращение длины $\Delta S_2 = S_2 - S_1 = 15$. Полученный маршрут оказался длиннее. Брать этот маршрут в качестве очередного в процедуре поиска минимума или не брать, зависит от вероятности перехода, вычисленной по формуле

$$P_* = T_1 e^{-\frac{\Delta S_2}{T_2}}$$

и от выпавшего в генераторе случайных чисел значения P_2 (значения P_1 оказалось в решении не задействовано, на первом этапе полученный маршрут уменьшится). Так как после первого этапа температура уменьшилась $T_1 = 50$, то

$$P_* = 100 e^{-\frac{15}{50}} = 74,08 > 62 = P_2$$

Выпавшее в генераторе случайных чисел P_2 попало в требуемый диапазон $[0, 74.08]$, следовательно, маршрут V_2 принимается в качестве очередного, несмотря на то, что длина больше.

1. Очередная замена элементов списка маршрута дает следующую последовательность $V_3 = [1, 4, 5, 2, 3, 6, 1]$ (рис. 5).

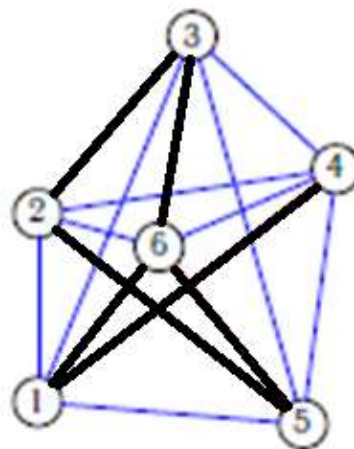


Рис. 5. Маршрут V_3

Находим длину маршрута.

$$S_3 = L_{1,4} + L_{4,5} + L_{5,2} + L_{2,3} + L_{3,6} + L_{6,1} = 39 + 26 + 35 + 24 + 22 + 20 = 166$$

Маршрут опять удлинился: $\Delta S_3 = S_3 - S_2 = 5$. Вычисляем верхнюю границу требуемого диапазона при температуре $T_3 = 0,5T_2 = 25$.

$$P_* = 100e^{-\frac{5}{25}} = 81,87 > 83 = P_3$$

Генератор случайных чисел дал число $P_3 = 83$, не попадающее в необходимый диапазон, следовательно, последний диапазон не принимается в качестве очередного и алгоритм возвращается к предыдущему маршруту V_2 .

1. Замена элементов 6 и 2 списка маршрута V_2 дает следующую последовательность $V_4 = [1, 6, 3, 2, 5, 4, 1]$ (рис.6).

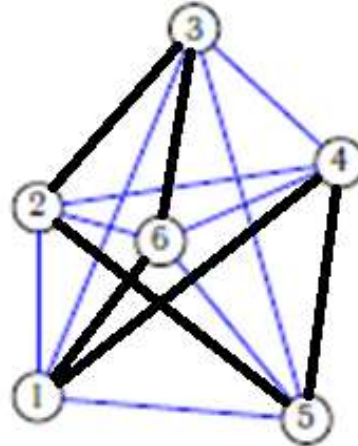


Рис. 6. Маршрут V_4

$$S_4 = L_{1,6} + L_{6,3} + L_{3,2} + L_{2,5} + L_{5,4} + L_{4,1} = 20 + 22 + 24 + 35 + 26 + 39 = 166$$

Маршрут опять удлинился: $\Delta S_4 = S_4 - S_2 = 166 - 161 = 5$. Уменьшаем температуру $T_4 = 0,5T_3 = 12,5$ и вычисляем верхнюю границу для допустимой вероятности

$$P_* = 100e^{-\frac{5}{12,5}} = 67,03 > 71 = P_4$$

Маршрут V_4 не принят. Таким образом, после четырех циклов алгоритма выбран маршрут V_2 длиной 161.

Общие схемы отжига.

Больцмановский отжиг

Исторически первой схемой метода отжига является так называемая схема Больцмановского отжига. В Больцмановском отжиге изменение температуры задается формулой:

$$T(k) = \frac{T_0}{\ln(1 + k)}, k > 0$$

Семейство распределений $\mathcal{G}(x, T)$ выбирается как семейство нормальных распределений с математическим ожиданием x и дисперсией T , т.е. задается плотностью

$$g(x'; x, T) = (2\pi T)^{-D/2} \exp(-|x' - x|^2 / (2T))$$

где D – размерность пространства состояний.

Пространство состояний предполагается метрическим. Для Больцмановской схемы доказано (рис.7), что при достаточно больших и общем количестве шагов, выбор такого семейства распределений гарантирует нахождение глобального минимума.

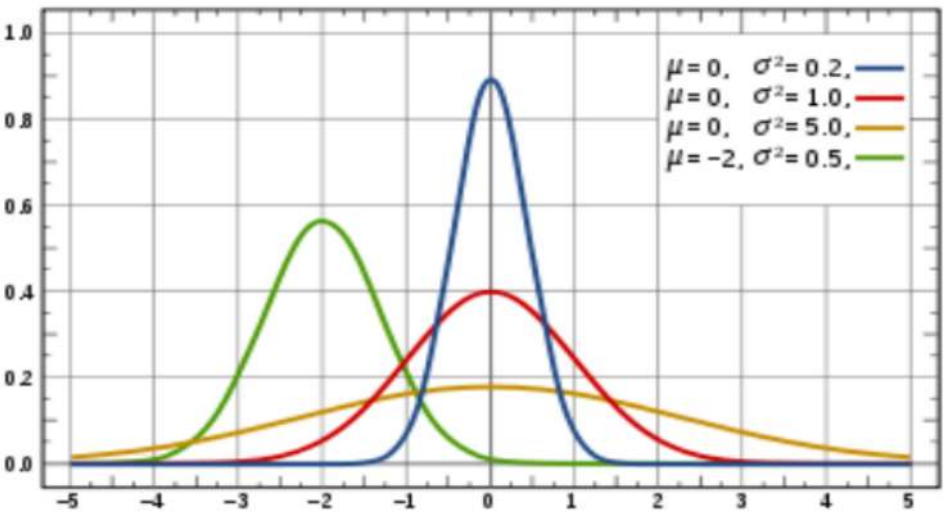


Рис. 7. Плотность нормального распределения

Отжиг Коши (быстрый отжиг)

Основным недостатком Больцмановского отжига является медленное убывание температуры. Например, для того чтобы понизить исходную температуру в 40 раз, требуется $e^{40} \approx 2,35 \times 10^{17}$ итераций, что уж вряд ли приемлемо для решения каких-либо задач. Ввиду этого Цу и Хартли предложили алгоритм, позволяющий использовать для изменения температуры схему

$$T(k) = \frac{T_0}{k}; k > 0$$

без потери гарантии нахождения глобального минимума. Это достигается за счет использования в качестве \mathcal{G} распределений Коши с плотностью

$$g(x'; x, T) = \frac{T}{(|x' - x|^2 + T^2)^{(D+1)/2}}$$

соответствующим образом нормированных. В случае $D = 1$ приходим к плотности

$$g(x'; x, T) = \frac{1}{\pi} \frac{T}{|x' - x|^2 + T^2}$$

К сожалению, это распределение не очень удобно моделировать в пространстве размерности больше 1. Этого можно избежать, например, с помощью перемножения D одномерных распределений Коши:

$$g(x'; x, T) = \frac{1}{\pi^D} \prod_{i=1}^D \frac{T}{|x' - x|^2 + T^2}$$

но в этом случае нахождении глобального минимума гарантируется только при законе изменения температуры не быстрее чем:

$$T(k) = \frac{T_0}{k^{1/D}}$$

что гораздо медленнее схемы $T(k) = \frac{T_0}{k}$

Сверхбыстрый отжиг

Недостатки двух предыдущих методов привели к тому, что в 1989 году американским исследователем Л. Ингбером был разработан метод сверхбыстрого отжига. В нем пространство S считается состоящим из D -мерных векторов $\begin{pmatrix} x_1 \\ \dots \\ x_D \end{pmatrix}$, где $x_i \in [A_i, B_i]$. Кроме этого, температура по каждой из координат может различаться, таким образом, T также является вектором размерности D .

Семейство распределений строится следующим образом. Вводится функция

$$g_T(y) = \prod_{i=1}^D \frac{1}{2(|y_i| + T_i) \ln(1 + 1/T_i)} \equiv \prod_{i=1}^D g_{(i;T)}(y_i), \quad y_i \in [-1, 1]$$

В качестве y для получения плотности распределений \mathcal{G} используется $\Delta x / B_i - A_i$, таким образом, новое значение x'_i вычисляется по формуле $x'_i = x_i + z_i(B_i - A_i)$, где z_i - случайная величина с плотностью $g_{(i;T)}$ на $[-1; 1]$. При этом выходящие за границы интервала значения параметра генерируются заново (пока не произойдет попадание в интервал) или приравниваются соответствующим границам.

Такую случайную величину легко промоделировать:

$$z_i = \operatorname{sgn}(\alpha_i - \frac{1}{2}) T_i \left(\left(1 + \frac{1}{T_i} \right)^{|2\alpha_i - 1|} - 1 \right)$$

где α_i - независимые случайные величины, распределенные равномерно на $[0, 1]$.

Доказано, что закон изменения температуры

$$T(k) = T_{(i;0)} \exp\left(-c_i k^{\frac{1}{D}}\right), \quad c_i > 0$$

дает статистическую гарантию нахождения глобального минимума. Для вероятности принятия также используется отдельная шкала температуры, изменяющаяся по такому же закону. Как правило, при реализации этого метода c_i управляется двумя параметрами:

$$c_i = m_i \exp(-n_i/D)$$

Алгоритм Ксин Яо

Алгоритм Ксин Яо был повторным применением идеи предыдущего алгоритма. В качестве $g_T(y)$ выбирается

$$g_T(y) = \prod_{i=1}^D g_{(i;T)}(y_i) = \prod_{i=1}^D \frac{1}{2(|y_i| + 1/\ln(1/T_i)) \ln(1 + \ln(1/T_i))}$$

Утверждается, что при изменении температуры по закону

$$T(k) = T_{(i;0)} \exp\left(-\exp\left(b_i k^{\frac{1}{D}}\right)\right), \quad b_i > 0$$

достигается статистическая гарантия нахождения глобального минимума.

Однако, как показано, увеличение скорости убывания температуры вовсе не означает ускорения в решении задачи. Более того, «размазанность» распределения приводит к тому, что метод генерирует огромное число «длинных» переходов, которые отвергаются в силу низкой вероятности их принятия.

Таким образом, несмотря на то, что этот процесс можно итерировать до бесконечности, получая законы изменения температуры вроде

$$T(k) = T_{(i;0)} \exp\left(-\exp\left(\exp\left(\dots \exp\left(b_i k^{\frac{1}{D}}\right)_i \dots\right)\right)\right)$$

ценность таких «улучшений» представляется сомнительной. Более того, легко видеть, что в пределе это приводит к тривиальному методу случайного поиска, которым является метод отжига при $T = 0$.

Это в небольшой степени применимо и к методу сверхбыстрого отжига, так что вопрос о скорости сходимости этих методов, а также о других методах, обеспечивающих не такое быстрое убывание температуры, но большую скорость сходимости, остается открытым. Вполне возможны задачи, на которых вторая итерация вышеописанного процесса может давать не плохие результаты.

Методы «тушения»

Далеко не всегда хватает вычислительных ресурсов на поиск глобального минимума. Кроме того, зачастую достаточно достигнуть не глобального оптимального решения задачи, а достаточно близкого к нему. Методы «тушения» не гарантируют нахождения глобального минимума, но, как правило, быстро находят близкое решение, а на практике зачастую и сам оптимум.

Основная идея этих методов заключается в том, чтобы скомбинировать семейство распределений \mathcal{G} одного из предыдущих четырех методов с более быстрым законом убывания температуры.

Например, можно рассматривать нормальное распределение \mathcal{G} из Больцмановского отжига, но при этом уменьшать температуру по закону

$$T_{k+1} = cT_k .$$

Как правило, в этом случае c выбирается между 0.7 и 0.99. Такой метод очень быстро сходится, и для конкретных задач может давать весьма неплохое решение, близкое к оптимальному, в условиях реального времени. Зачастую они основаны либо на нормальном распределении, либо на распределении для сверхбыстрого отжига. Кроме того, встречаются специальные распределения, подобранные опытным путем для решения конкретных задач.

Масштабирование в ходе отжига

Зачастую при реализации сверхбыстрого отжига с большой размерностью используются технология масштабирования отжига, иногда также применяемая и к другим вариантам отжига. При использовании этой технологии периодически во время отжига производится следующая операция.

Обозначим за s_i значение некоторой оценки производной целевой функции по i -й координате в точке текущего минимума:

$$s_i \geq \left| \frac{\partial f}{\partial x_i} (x_i^{min}) \right|$$

Кроме того, пусть $s_{max} = \max_{1 \leq i \leq D} s_i$. После этого шага (называемым в этом алгоритме временем отжига) и температура для каждой размерности изменяются следующим образом:

$$\begin{aligned} T_i' &= T_i(k_i)(s_{max}/s_i) \\ k_i' &= (\ln(T_{(i,0)}/T_i')/s_i)^D \end{aligned}$$

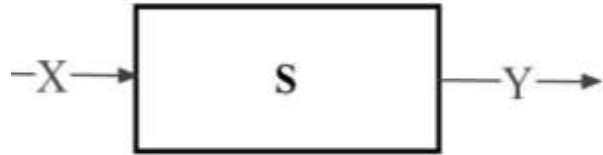
Таким образом, значение $T_{(i,0)}$ сохраняются, а время отжига (которое теперь может принимать не только целые значения) масштабируется согласно температуре.

СИСТЕМА И ЕЕ СВОЙСТВА

Статические свойства систем

Целостность.

Открытость.



**Внутренняя неоднородность систем,
различимость частей.**

Структурированность.

Динамические свойства систем

Функциональность.

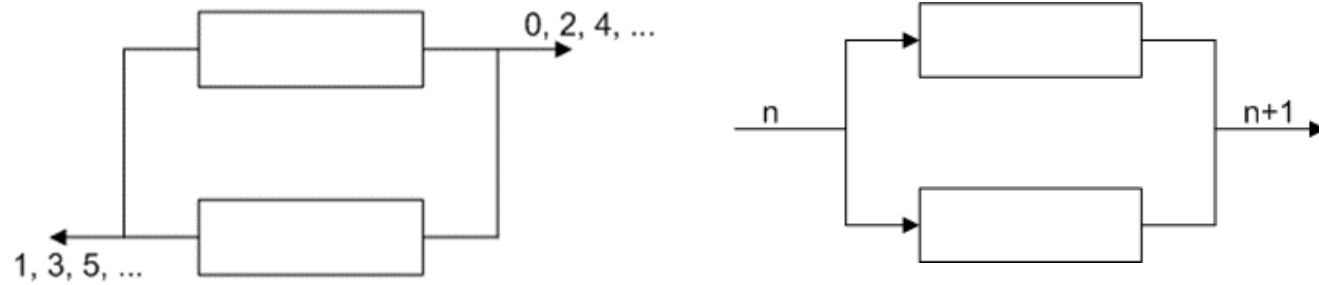
Стимулируемость.

Изменчивость системы со временем.

Существование в изменяющейся среде.

Синтетические свойства систем

Эмерджентность.



Неразделимость на части.

Ингерентность.

Целесообразность.

