В данной статье приняты следующие обозначения:

1. *n* - длинна вектора;
2. *k* - разрядность фрагментов;
3. *m = n / k* - количество фрагментов.

Для описания работы алгоритма необходимо ввести следующие термины и структуры:

* Функциональный преобразователь (ФП) - преобразователь *fi* *(i=1,2..2m-1)*, который каждому значению *x* из множества *{0,1..2k-1}* сопоставляет значение *y* из множества *{0,1..2k-1}*: *y = fi(x)*.
* Ячейка - хранит числовое значение из множества {0,1..2k-1} или пустое значение.
* Блок - комбинация двух ячеек: *in* и *out*.
* Снимок - множество ячеек, хранящих суммарную информацию о числовом значении всех звений системы, которые в совокупности составляют маршрут, по которому вектор *x*, пройдя по системе, превращается в вектор *y*.
* Шаг - характеризуется координатами, множеством действий, к которым привёл этот шаг, а также множеством использованных чисел. Множество использованных чисел содержит числа, для которых были попытки сделать этот шаг. Также имеет логику, определяющую, не был ли достигнут лимит разрешенного количества попыток. Решение о выполнении шага принимает алгоритм.
* Действие - характеризуется координатами и числом. В отличие от шага, действие является безусловным, то есть выполняется однозначно в силу текущих значений в снимке как следствие выполнение шага или другого действия.
* Конфликт - ситуация в снимке, когда выполнение какого-то действия невозможно в силу текущего состояния снимка (совокупности всех предыдущих действий в нём), то есть когда выполнение действия нарушит математические связи между узлами системы.

Снимок имеет следующую структуру:

1. *m* ячеек для значений *x11, x12 ... x1m*, соответственно;
2. для каждого слоя *i (i=1..m-1)* и колонки в нём *j (j=1..2m-i)*: блок, показывающий, какие значения *in* и *out* используются для ФП *fi* в *i*-м слое *j*-й колонки.
3. *m* ячеек для значений *y1, y2 ... ym*, соответственно.

На вход алгоритм получает целевой вектор *y*.Перед началом работы алгоритма необходимо создать пустое множество *xs*, куда будут помещаться найденные векторы *x*.

1. Создать новый снимок, инициализировать его ячейки *y1, y2 ... ym* из вектора *y*.
2. Найти (случайным образом) пустую ячейку в снимке. Если таковой нету - перейти к Пункту 8, иначе создать новый шаг, инициализировать его координаты координатами найденной ячейки.
3. Выбрать (случайным образом) неиспользованное ранее для текущего шага число на основе его множества использованных чисел, добавить это число в его множество использованных чисел. Если не удалось выбрать число (было достигнут лимит допустимого количества попыток) - перейти к Пункту 6.
4. Получить множество действий, к которым приводит выполнение текущего шага с полученным числом. Если в процессе получения действий в снимке возник конфликт - перейти на Пункт 6.
5. Применить полученные в Пункте 4 действия и сохранить их в текущем шаге, положить в стек текущий шаг, установить текущий шаг равным *null* и перейти к Пункту 7.
6. Пока был достигнут лимит количества попыток для текущего шага: если стек пустой (система насыщена, не удалось найти очередной вектор) - перейти к Пункту 10, иначе достать шаг из стека и присвоить текущему шагу; откатить все действия для шага.
7. Если шаг равен *null* - перейти к Пункту 2, иначе перейти к Пункту 3.
8. Снимок полностью заполнен. Если полученный *x* уже присутствует во множестве *xs* - перейти к шагу 10, иначе добавить *x* в множество *xs*.
9. Перейти на Пункт 1.
10. Вывести результат работы: множество x*s*.