

МВ И ССО РЕСПУБЛИКИ УЗБЕКИСТАН

АНДИЖАНСКИЙ МАШИНАСТРОИТЕЛЬНЫЙ
ИНСТИТУТ

КАФЕДРА «АВТОМАТИЗАЦИЯ
МАШИНОСТРОИТЕЛЬНОГО ПРОИЗВОДСТВА»

САМОСТОЯТЕЛЬНАЯ РАБОТА

ПО ДИСЦИПЛИНЕ

«ТАУ»

На тему: Решения задач оптимизации

нормализация критериев.

ВЫПОЛНИЛ(А): Махсудова М

ПРИНЯЛ(А): Сабиров У

Андижан -2016

Решения задач оптимизации нормализация критериев.

План.

1. Нормализация критериев.
2. Парето-оптимальных решений.
3. Скалярные методы решения задачи многокритериальной оптимизации.

Список литературы

Как правило, В большинстве задач многокритериальной оптимизации критерии имеют существенно различный физический (экономический) смысл, разные единицы измерения, поэтому сравнение критериев по численному значению невозможно, что в значительной степени затрудняет применение всех методик многокритериальной оптимизации.

Поэтому в большинстве случаев в решение многокритериальной задачи требуется включение предварительного этапа - нормализации критериев.

Нормализация критериев представляет собой однозначное отображение функции $f_k(X), \forall k \in K$, из R^N в R^N . Для нормализации критериев в векторных задачах используется линейное преобразование:

$$f_k(X) = af'(X) + c, \forall k \in K$$

или

$$f_k(X) = (f_k(X) + c) / a, \forall k \in K,$$

где $f_k(X), \forall k \in K$ - первоначальное значение критерия; $f_k(X), \forall k \in K$ - нормализованное значение.

Такая нормализация критериев в оптимизационной задаче не влияет на результат решения. В случае, если решается выпуклая оптимизационная задача:

$$\max F(X), X \in S,$$

X

то в точке оптимума

$$X^* \in S, dF(X^*)/dX = 0.$$

Если же решается оптимизационная задача:

$$\max(aF(X) + c), X \in S,$$

то в точке оптимума

$$X^* \in S, d(aF(X^*) + c)/dX = 0,$$

т.е. результат идентичен.

К нормализации критериев в векторных задачах предъявляются два основных требования:

нормализованные критерии должны быть измерены в одних и тех же величинах;

в точках оптимума $X_k, V_k \in K$ величины всех критериев должны иметь одинаковую величину.

При выполнении этих требований представляется возможным сравнить критерии по их численному значению. В нашем случае - решении задачи многокритериальной оптимизации с максимумом векторной целевой функции применяется следующая нормализация критериев:

$$X_k(X) = (f_k(X) - f_{k\min}) / (f_{k\max} - f_{k\min}).$$

При этом относительная оценка $X_k(X)$, $k = 1, K$ на всем множестве допустимых точек лежит в пределах

0 При этом величины $1 - X_k(X)$ называются относительными отклонениями $X_k(X)$.

Следует заметить, что при всех достоинствах предварительной нормализации критериев у нее имеется важный недостаток (общий для всех относительных величин) - нормализованный критерий не всегда имеет столь однозначную экономическую интерпретацию, как обычный. Например, если относительная оценка критерия f_9 - норматив текущей ликвидности - равна 0,9 (т.е. 90 % от максимально возможной), то трудно сказать, достаточный ли это показатель для банка. Если же знать, что в абсолютных величинах данная величина составляет 94 % (при максимуме в 106 %), а требуемое инструкцией минимальное значение составляет 70 %, то на основании этого можно сделать более определенные выводы. Поэтому в ходе решения задачи многокритериальной оптимизации необходимо пользоваться параллельно обычными и нормализованными критериями. Методика построения множества

Парето-оптимальных решений.

Выделение Парето-оптимального подмножества - достаточно трудная задача, для которой пока не известен универсальный алгоритм. Однако для некоторых частных случаев такие алгоритмы известны, например, для случая линейной задачи векторной оптимизации существует обобщенный вариант симплекс-метода, который позволяет выделить все Парето-оптимальные

крайние точки и неограниченные эффективные ребра. Однако в нашем случае он не подходит, так как некоторые из целевых функций являются нелинейными. В этом случае напрашивается решение, использующее идею полного просмотра всех возможных вариантов решений и выбора из них лучшего. Однако, естественно, что такой полный просмотр невозможен, так как количество точек просмотра бесконечно. Для того, чтобы уменьшить количество просматриваемых точек можно (конечно, в ущерб получаемого объема информации) каким-либо разумным способом организовать процедуру просмотра. На этой идее и основан метод решения проектно-конструкторских задач с противоречивыми критериями, предложенный И. М. Соболев и Р. Б. Статниковым и основанный на утверждении, что максимальное число просматриваемых точек при минимуме вычислений достигается, если точки выбираются из так называемой ЛПт-последовательности. Название ЛПт-последовательность появилось как сокращение фразы

"бесконечные последовательности точек, любой двоичный участок которых есть Пт-сетка". Таким образом, просматривая варианты в точках, соответствующих ЛПт-последовательности и, вычисляя значения критериев в этих точках, можно принимать обоснованные решения. Для выделения среди полученных точек Парето-оптимальных используется метод, основанный на применении алгоритмов сортировки. Достоинством данного метода является отсутствие необходимости решения скалярных задач оптимизации, которые в некоторых случаях (сложные нелинейные функции с обилием разрывов и т. д.) требуют значительных затрат машинного времени. В нашем случае это не столь актуально, как в инженерных задачах, так как экстремумы всех использованных функций легко вычисляются.

Поэтому в качестве метода построения множества Парето был выбран приближенный графоаналитический метод Н. Н. Моисеева, который заключается в последовательном итеративном процессе решения простейших оптимизационных задач. В случае двух критериев он применяется следующим образом: сначала задаются начальными произвольными значениями критериев: $f_1 = c_1$; $f_2 = c_2$. Затем решаются две оптимизационные задач:

1) $\max f_1, f = C_2$; 2) $\max f_2, f = c_1$

Решив эти две задачи находят точки a и b . Прямая, соединяющая эти две точки является областью Парето в первом приближении. Далее решаются две

аналогичные задачи. При этом задаются значениями критериев: $f = c_3$; $f_2 = c_4$. Затем решаются две оптимизационные задачи:

1) $\max f_1, f = C_4$; 2) $\max f_2, f = C_3$.

Через полученные точки снова проводят прямые. После соединения точек c и d получают ломаную $acdb$, которая является областью Парето второго приближения. В большинстве случаев второе приближение является достаточным. В случае большего числа критериев метод теряет наглядность, но может выполняться с помощью ЭВМ. Важной проблемой является визуальное представление множества Парето. В случае двух критериев оно отображается линией на плоскости. В случае трех критериев множество Парето можно визуализировать поверхностью в пространстве, однако, в отличие от линии на плоскости сделать выбор на основании трехмерного графика достаточно трудно. Поэтому возможно применение альтернативного метода линий уровня, который можно обобщить на большее число критериев.

При построении множества Парето-оптимальных решений ввиду трудностей многомерной визуализации необходимо выбрать два - три критерия, наиболее актуальных в данный момент.

Скалярные методы решения задачи многокритериальной оптимизации.

Методика построения множества Парето позволяет представить наиболее полную информацию о многокритериальной задаче оптимизации лицу, принимающему решение, без каких-либо предпосылок об относительной важности критериев. Однако данный метод достаточно громоздок и трудно применим в случае большого числа критериев (более трех). Поэтому на практике более широко применяются методы, позволяющие тем или иным способом привести решение задачи векторной оптимизации к скалярной.

Метод главного критерия

Наиболее простым и часто применяющимся методом является выделение одного критерия в качестве главного и перевод остальных критериев в разряд ограничений путем формулировки дополнительных ограничений на значения этих критериев. Конкретные значения данных дополнительных ограничений могут быть установлены, например, с помощью статистических методов, или экспертным путем на основании

неформальных соображений. Вероятно, в большинстве ситуаций для банка основным критерием, подлежащим максимизации, будет являться показатель прибыльности (рентабельности), а на критерии, отражающие различные аспекты надежности, будут накладываться ограничения, так как конечной целью деятельности всякой коммерческой организации является получение прибыли, а увеличение надежности путем, например, чрезмерного повышения ликвидности выше разумных пределов не только не увеличит доверие клиентов и партнеров, но и может поставить под угрозу будущее состояние банка ввиду снижения прибыльности. Однако в некоторых ситуациях оправдано сосредоточение усилий на максимизации какого-либо другого критерия, помимо прибыльности. Так, в предвидении кризиса доверия к банковской системе разумной является максимизация ликвидности с целью удовлетворения массовых требований по возврату вкладов. В случае, если банк желает расширить свою ресурсную базу путем привлечения вкладов такой группы населения и хозяйственных агентов, которая при выборе банка ориентируется на его место в популярных банковских рейтингах, то следует максимизировать итоговый рейтинг банка по модели Кромонава, как наиболее популярной. В целом, метод приведения многокритериальной задачи к однокритериальной путем наложения дополнительных ограничений на менее важные критерии является широко распространенным ввиду своей понятности, простоты интерпретации результатов и невысоких требований к математической подготовке эксперта, программному обеспечению и быстродействию ЭВМ. Однако данному методу присущ ряд фундаментальных недостатков. Прежде всего, данный метод значительно упрощает структуру исходной задачи, не учитывает разницу в значениях критериев, переведенных в разряд ограничений. Кроме того, достаточно трудной задачей является формулирование ограничений на значения менее важных критериев. Если задать слишком низкие ограничения, то полученная точка не обязательно будет Парето-оптимальной (в случае, если целевая функция имеет несколько экстремумов), а если слишком высокие, то значение целевой функции (главного критерия) в полученной точке будет слишком низким по сравнению с его абсолютно достижимым максимумом (без учета ограничений на другие критерии). Однако данный метод используется при построении Парето-оптимального множества по методике академика Моисеева.

Метод уступок

Улучшенной разновидностью метода перевода менее важных критериев в ограничения является метод последовательных уступок (называемый также методом оптимизации по последовательно применяемым критериям), предлагаемый прежде всего В. В. Подиновским в ряде работ. Его суть состоит в следующем. Проводится анализ относительной важности критериев и критерии располагаются и нумеруются в порядке убывания важности. Производится оптимизация по первому критерию и определяется его наибольшее значение f^* . Далее эксперт оценивает величину допустимого снижения (уступки) данного критерия ($f_1 - A/1$) и ищется оптимум второго по важности критерия и т.д. После оптимизации последнего по важности критерия при условии, что значение каждого критерия $k = 1, K$ должно быть не меньше ($f_1 - A_k$), $k = 1, K$, получаемые решения считаются оптимальными.

Достоинства данного метода в его простоте и наглядности. Важным преимуществом является возможность целенаправленного участия лица, принимающего решение в процессе оптимизации с учетом ранее полученных (на предыдущем этапе оптимизации) данных путем выбора величины уступки по каждому критерию.

Основным теоретическим недостатком данного метода является то, что на каждом шаге происходит усечение множества точек, оптимальных по Парето, отсюда в общем случае получившееся решение не оптимально по Парето, т. е. требуется дополнительное доказательство оптимальности по Парето данного решения, что является очень сложной процедурой. Иначе придется смириться с тем, что данное решение, хотя и удовлетворяет лицу, принимающее решение значениями всех критериев, но не обязательно является оптимальным. Однако на практике это не столь важно, так как в реальной ситуации ищут, как правило, не оптимальное, но «достаточно хорошее» решение. Вторым недостатком является сложность выбора и обоснования величин уступок по отдельным критериям, так как величины уступок не соизмеримы между собой ввиду различной экономической сущности разных критериев. Однако от второго недостатка можно избавиться применением нормализации критериев.

Свертка критериев

Другая очень распространенная группа методов скаляризации векторной задачи математического программирования - свертка критериев.

Существует большое количество разных видов сверток [11, 12]. Теоретически все они базируются на подходе, связанном с понятием функции полезности лица, принимающего решение. При данном подходе предполагается, что лицо, принимающее решение, всегда имеет функцию полезности, независимо от того, может ли лицо, принимающее решение задать ее в явном виде (т.е. дать ее математическое описание). Эта функция отображает векторы критериев на действительную прямую так, что большее значение на этой прямой соответствует более предпочтительному вектору критериев. Смысл разных сверток состоит в том, чтобы из нескольких критериев получить один «коэффициент качества» (сводный критерий), приближенно моделируя таким образом неизвестную (не заданную в явном виде) функцию полезности лица, принимающего решение. Наиболее популярной сверткой является метод взвешенных сумм с точечным оцениванием весов. При этом задается вектор весовых коэффициентов критериев, характеризующий относительную важность того или иного критерия:

$$A = \{a_k, k = 1 \dots K\}.$$

Весовые коэффициенты обычно используются в нормированном виде и удовлетворяют равенству:

K

$$\sum_{k=1}^K a_k = 1, a_k > 0, \forall k \in K,$$

$k=1$

т.е. предполагается, что весовые коэффициенты неотрицательны. Каждый критерий умножается на свой весовой коэффициент, а затем все взвешенные критерии суммируются и образуют взвешенную целевую функцию, значение которой интерпретируются как «коэффициент качества» полученного решения. Полученная скаляризованная функция максимизируется на допустимой области ограничений.

Получается однокритериальная (скалярная) задача математического программирования:

$$F_0 = \max_{X} \sum_{k=1}^K a_k f_k(X).$$

$k = 1$

В результате решения данной задачи получается точка оптимума X_0 .

Основным достоинством данной свертки является то, что с ней связаны классические достаточные и необходимые условия оптимальности по Парето (теоремы Карлина).

Теорема Карлина 1.

В выпуклой задаче многокритериальной оптимизации точка $X_0 \in S$ оптимальна по Парето, если существует вектор весовых коэффициентов $A_0 = \{a^k > 0, k = 1, K\}$, для которого выполняется соотношение:

$$X_{a_0} / (X_0) = \max_{X \in S} \sum_{k=1}^K a^k f_k(X).$$

$$a^k, k = 1, K$$

$$\sum_{k=1}^K a^k = 1$$

Теорема Карлина 2.

Если в выпуклой задаче многокритериальной оптимизации точка $X_0 \in S$ Парето-оптимальна, то существует вектор весовых коэффициентов $A_0 = \{a^k > 0, k = 1, K\}$, для которого выполняется соотношение:

$$X_{a_0} / (X_0) = \max_{X \in S} \sum_{k=1}^K a^k f_k(X). \quad (68)$$

$$a^k, k = 1, K \quad \sum_{k=1}^K a^k = 1$$

$$\sum_{k=1}^K a^k = 1$$

Согласно данным теоремам, данную свертку можно использовать для получения Парето-оптимальных точек.

Примером данной свертки может служить итоговый рейтинг надежности банка Кромонава, полученный как аддитивная свертка ряда коэффициентов. Достоинством данного метода является то, что он согласно теореме Карлина генерирует Парето-оптимальные точки. Однако ему присущ целый ряд фундаментальных недостатков. Во-первых, неявная функция полезности лица, принимающего решения, как правило, нелинейна, поэтому «истинные» веса критериев (т.е. такие веса, при которых градиент взвешенное целевой функции совпадает по направлению в градиентом функции полезности) будут меняться от точки к точке, поэтому можно говорить лишь о локально подходящих весах, кроме того, часто лицо, принимающее решение вообще не может задать весовые коэффициенты. Во-вторых, далеко не всегда потеря качества по одному из критериев компенсируется приращением качества по другому. Поэтому полученное

решение, оптимальное в смысле единого суммарного критерия, может характеризоваться низким качеством по ряду частных критериев и быть поэтому абсолютно неприемлемым. В-третьих, полученное решение часто бывает неустойчиво, т. е. малым приращениям весовых коэффициентов соответствуют большие приращения целевых функций. В-четвертых, свертка критериев разной физической природы не позволяет интерпретировать значение взвешенной целевой функции. В-пятых, значительные затруднения могут возникнуть в случае сильной корреляции между критериями.

Некоторые из вышеперечисленных недостатков могут быть скорректированы. Так, в случае разной физической (экономической) природы критериев возможна их нормализация и последующая свертка нормализованных критериев. Чтобы исключить неприемлемо низкие значения отдельных критериев, можно наложить дополнительные ограничения на эти критерии.

Другим методом борьбы с данным недостатком - неприемлемо низкими значениями отдельных критериев при хорошем значении суммарного критерия - является применение сверток не аддитивного, а мультипликативного вида:

$$G_0 = \max_{\Pi} (a_k/k(X)) P_k. \quad (69)$$

кеК

Однако данная свертка не получила большого распространения ввиду того, что существуют аналогичные, но более перспективные виды сверток.

(70)

Так, существует свертка вида:

$$\hat{\pi} = X | /k - /k(X)$$

/к

Наиболее широкое применение данная свертка получила при $p = 2$, которая трактуется как минимизация суммы квадратов относительных отклонений функционалов от своих достижимых оптимальных значений. Данная точка в случае равноценности критериев показывает решение, наиболее близкое к недостижимой «идеальной» точке (в которой все критерии принимают свое максимальное значение). Однако данной свертке также свойственен следующий распространенный недостаток: «хорошее»

значение сводного критерия достигается ценой низких значений некоторых частных критериев.

Методики, основанные на гарантированном результате Данный недостаток отсутствует в методиках, основанных на гарантированном результате (максимине, минимаксе). Данный принцип впервые был предложен Карлиным [19] в следующей постановке:

$$\text{тахтіп}^{\wedge}(X) = \{ /k, k = 1K \}. (71)$$

X_k

Данная задача называется максимизацией минимальной компоненты. Но, так как критерии часто измеряются в разных единицах, то не представляется возможным сравнивать критерии между собой и вести совместную оптимизацию.

$k = 1$

Машуниным был предложен усовершенствованный вариант данной методики, основанный на использовании нормализации критериев.

Машунин вводит понятие уровня X -нижней из относительных оценок:

$$X = \text{тіп } X_k (X) (72)$$

и преобразует максиминную задачу

$$X_0 = \text{тахтіп } X_k (X) (73)$$

в экстремальную задачу

$$X_0 = \text{тах } X, X < X_k (X), k = 1, K. (74)$$

Данная задача является формализованным представлением принципа максимальной эффективности.

Методика, основанная на принципе максимина, позволяет оценить расположение условного центра многомерного множества

Парето. Применение данного метода полезно даже в условиях задачи с двумя или тремя критериями, когда возможна визуализация

множества Парето, так как она дает дополнительную информацию о возможностях компромисса между критериями.

2.Практическая часть.

17-Variant

>>

$T1=0.44, Tm=0.242, Tya=0.02, K1=2.5, Kd=0.91, Kf=0.71, Ktg=0.36, Keu=10, Mc=45$
 $, \omega_{gad}=200, \delta\omega_{gad}=0.8, p=3, \omega=(0:0.17:6)$

$T1 =$

0.4400

$Tm =$

0.2420

$Tya =$

0.0200

$K1 =$

2.5000

$Kd =$

0.9100

$Kf =$

0.7100

$Ktg =$

0.3600

$Keu =$

10

Mc =

45

omegad =

200

deltaomegad =

0.8000

p =

3

omega =

Columns 1 through 8

0 0.1700 0.3400 0.5100 0.6800 0.8500 1.0200 1.1900

Columns 9 through 16

1.3600 1.5300 1.7000 1.8700 2.0400 2.2100 2.3800 2.5500

Columns 17 through 24

2.7200 2.8900 3.0600 3.2300 3.4000 3.5700 3.7400 3.9100

Columns 25 through 32

4.0800 4.2500 4.4200 4.5900 4.7600 4.9300 5.1000 5.2700

Columns 33 through 36

5.4400 5.6100 5.7800 5.9500

>> C=62.7*10^-6,R4=1.11*10^3,R1=10^4

C =

6.2700e-005

R4 =

1110

R1 =

10000

```
>> Wp1=(Tya*p+1)
Aomega1=sqrt((Tya^2).*(omega.^2)+1)
phiomega1=atan(Tya.*omega)
```

```
Wp2=1/(Tm*p+1)
Aomega2=1./sqrt((Tm^2).*(omega.^2)+1)
phiomega2=-atan(Tm.*omega)
```

```
k0=(((Kf*Mc)/deltaomegad)-1)/(Keu*K1*Kd*Ktg)
kpc=k0*Keu*K1*Kd*Ktg
Wp3=kpc/(T1*p+1)
Aomega3=kpc./sqrt((T1^2).*(omega.^2)+1)
phiomega3=-atan(T1.*omega)
```

```
W1p=(0.1*p+1)
A1omega=sqrt(0.01.*(omega.^2)+1)
phi1omega=atan(0.1.*omega)
```

```
W2p=1/(10*p+1)
A2omega=1./sqrt(100.*(omega.^2)+1)
phi2omega=-atan(10.*omega)
```

```
W3p=10/(1*p+1)
A3omega=10./sqrt(1.*(omega.^2)+1)
phi3omega=-atan(1.*omega)
```

Wp1 =

1.0600

Aomega1 =

Columns 1 through 8

1.0000 1.0000 1.0000 1.0001 1.0001 1.0001 1.0002 1.0003

Columns 9 through 16

1.0004 1.0005 1.0006 1.0007 1.0008 1.0010 1.0011 1.0013

Columns 17 through 24

1.0015 1.0017 1.0019 1.0021 1.0023 1.0025 1.0028 1.0031

Columns 25 through 32

1.0033 1.0036 1.0039 1.0042 1.0045 1.0048 1.0052 1.0055

Columns 33 through 36

1.0059 1.0063 1.0067 1.0071

phiomega1 =

Columns 1 through 8

0 0.0034 0.0068 0.0102 0.0136 0.0170 0.0204 0.0238

Columns 9 through 16

0.0272 0.0306 0.0340 0.0374 0.0408 0.0442 0.0476 0.0510

Columns 17 through 24

0.0543 0.0577 0.0611 0.0645 0.0679 0.0713 0.0747 0.0780

Columns 25 through 32

0.0814 0.0848 0.0882 0.0915 0.0949 0.0983 0.1016 0.1050

Columns 33 through 36

0.1084 0.1117 0.1151 0.1184

Wp2 =

0.5794

Aomega2 =

Columns 1 through 8

1.0000 0.9992 0.9966 0.9925 0.9867 0.9795 0.9709 0.9609

Columns 9 through 16

0.9499 0.9378 0.9248 0.9111 0.8967 0.8818 0.8665 0.8510

Columns 17 through 24

0.8353 0.8195 0.8036 0.7879 0.7722 0.7567 0.7414 0.7264

Columns 25 through 32

0.7116 0.6971 0.6829 0.6691 0.6556 0.6424 0.6295 0.6170

Columns 33 through 36

0.6049 0.5931 0.5816 0.5704

phiomega2 =

Columns 1 through 8

0 -0.0411 -0.0821 -0.1228 -0.1631 -0.2029 -0.2420 -0.2804

Columns 9 through 16

-0.3180 -0.3546 -0.3903 -0.4250 -0.4586 -0.4911 -0.5226 -0.5529

Columns 17 through 24

-0.5821 -0.6103 -0.6374 -0.6635 -0.6885 -0.7125 -0.7356 -0.7578

Columns 25 through 32

-0.7790 -0.7994 -0.8190 -0.8378 -0.8559 -0.8732 -0.8898 -0.9058

Columns 33 through 36

-0.9212 -0.9359 -0.9501 -0.9638

k0 =

4.7543

kpc =

38.9375

Wp3 =

16.7834

Aomega3 =

Columns 1 through 8

38.9375 38.8290 38.5090 37.9927 37.3036 36.4703 35.5239 34.4950

Columns 9 through 16

33.4122 32.3002 31.1799 30.0678 28.9766 27.9155 26.8910 25.9072

Columns 17 through 24

24.9665 24.0696 23.2166 22.4067 21.6386 20.9105 20.2208 19.5674

Columns 25 through 32

18.9483 18.3616 17.8054 17.2778 16.7770 16.3014 15.8493 15.4193

Columns 33 through 36

15.0101 14.6202 14.2485 13.8939

phiomega3 =

Columns 1 through 8

0 -0.0747 -0.1485 -0.2207 -0.2907 -0.3579 -0.4219 -0.4823

Columns 9 through 16

-0.5392 -0.5925 -0.6422 -0.6885 -0.7315 -0.7714 -0.8084 -0.8428

Columns 17 through 24

-0.8747 -0.9044 -0.9320 -0.9576 -0.9816 -1.0039 -1.0247 -1.0443

Columns 25 through 32

-1.0626 -1.0797 -1.0959 -1.1110 -1.1253 -1.1388 -1.1516 -1.1636

Columns 33 through 36

-1.1751 -1.1859 -1.1962 -1.2059

W1p =

1.3000

A1omega =

Columns 1 through 8

1.0000 1.0001 1.0006 1.0013 1.0023 1.0036 1.0052 1.0071

Columns 9 through 16

1.0092 1.0116 1.0143 1.0173 1.0206 1.0241 1.0279 1.0320

Columns 17 through 24

1.0363 1.0409 1.0458 1.0509 1.0562 1.0618 1.0676 1.0737

Columns 25 through 32

1.0800 1.0866 1.0933 1.1003 1.1075 1.1149 1.1225 1.1304

Columns 33 through 36

1.1384 1.1466 1.1550 1.1636

phi_{omega} =

Columns 1 through 8

0 0.0170 0.0340 0.0510 0.0679 0.0848 0.1016 0.1184

Columns 9 through 16

0.1352 0.1518 0.1684 0.1849 0.2012 0.2175 0.2337 0.2497

Columns 17 through 24

0.2656 0.2813 0.2970 0.3124 0.3277 0.3429 0.3579 0.3727

Columns 25 through 32

0.3874 0.4019 0.4162 0.4303 0.4443 0.4580 0.4716 0.4850

Columns 33 through 36

0.4982 0.5112 0.5241 0.5367

W_{2p} =

0.0323

A_{2omega} =

Columns 1 through 8

1.0000 0.5070 0.2822 0.1924 0.1455 0.1168 0.0976 0.0837

Columns 9 through 16

0.0733 0.0652 0.0587 0.0534 0.0490 0.0452 0.0420 0.0392

Columns 17 through 24

0.0367 0.0346 0.0327 0.0309 0.0294 0.0280 0.0267 0.0256

Columns 25 through 32

0.0245 0.0235 0.0226 0.0218 0.0210 0.0203 0.0196 0.0190

Columns 33 through 36

0.0184 0.0178 0.0173 0.0168

phi2omega =

Columns 1 through 8

0 -1.0391 -1.2847 -1.3772 -1.4248 -1.4537 -1.4731 -1.4870

Columns 9 through 16

-1.4974 -1.5055 -1.5120 -1.5174 -1.5218 -1.5256 -1.5288 -1.5316

Columns 17 through 24

-1.5340 -1.5362 -1.5381 -1.5398 -1.5414 -1.5428 -1.5441 -1.5452

Columns 25 through 32

-1.5463 -1.5473 -1.5482 -1.5490 -1.5498 -1.5505 -1.5512 -1.5518

Columns 33 through 36

-1.5524 -1.5530 -1.5535 -1.5540

W3p =

2.5000

A3omega =

Columns 1 through 8

10.0000 9.8586 9.4677 8.9084 8.2693 7.6194 7.0007 6.4334

Columns 9 through 16

5.9239 5.4710 5.0702 4.7157 4.4016 4.1225 3.8736 3.6509

Columns 17 through 24

3.4507 3.2700 3.1063 2.9575 2.8217 2.6973 2.5831 2.4778

Columns 25 through 32

2.3805 2.2904 2.2067 2.1287 2.0560 1.9879 1.9241 1.8643

Columns 33 through 36

1.8079 1.7549 1.7048 1.6574

phi3omega =

Columns 1 through 8

0 -0.1684 -0.3277 -0.4716 -0.5972 -0.7045 -0.7953 -0.8719

Columns 9 through 16

-0.9368 -0.9919 -1.0391 -1.0797 -1.1150 -1.1459 -1.1730 -1.1971

Columns 17 through 24

-1.2185 -1.2377 -1.2549 -1.2706 -1.2847 -1.2977 -1.3095 -1.3204

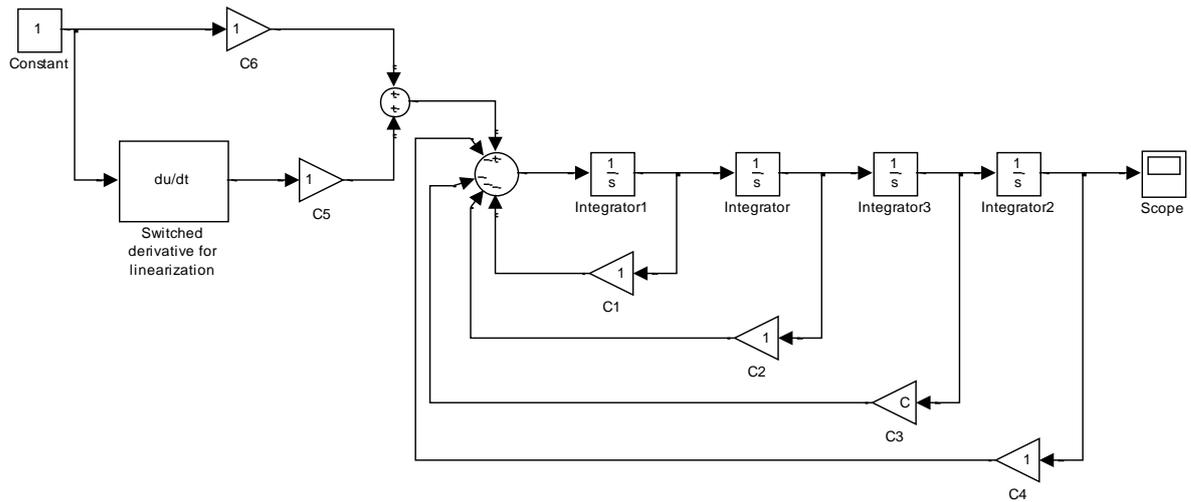
Columns 25 through 32

-1.3304 -1.3397 -1.3483 -1.3563 -1.3637 -1.3707 -1.3772 -1.3833

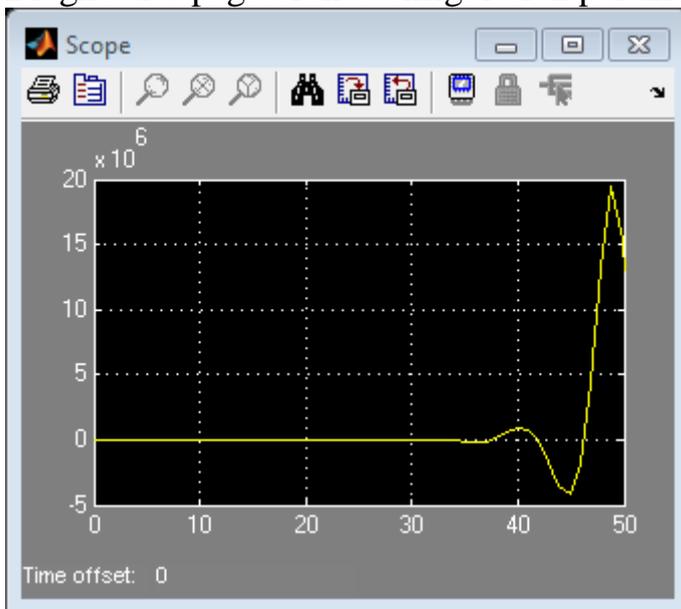
Columns 33 through 36

-1.3890 -1.3944 -1.3995 -1.4043

>>simulink



Warning: Using a default value of 1 for maximum step size. The simulation step size will be equal to or less than this value. You can disable this diagnostic by setting 'Automatic solver parameter selection' diagnostic to 'none' in the Diagnostics page of the configuration parameters dialog.



```
>> w1=tf([Kd],[0.003247 Tm 1])
w2=tf([K1],[T1 1])
w3=tf([Tya 1],[0.003247 Tm 1])
```

Transfer function:

0.91

 $0.003247 s^2 + 0.242 s + 1$

Transfer function:

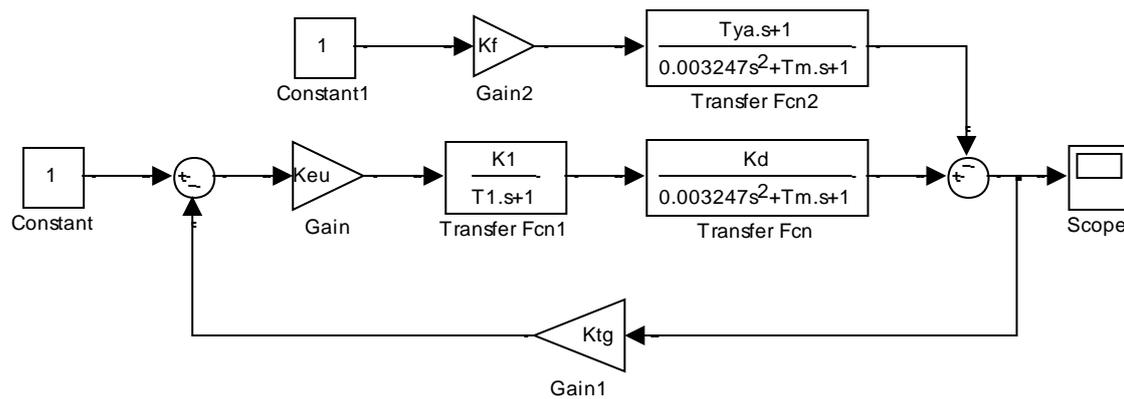
2.5

 $0.44 s + 1$

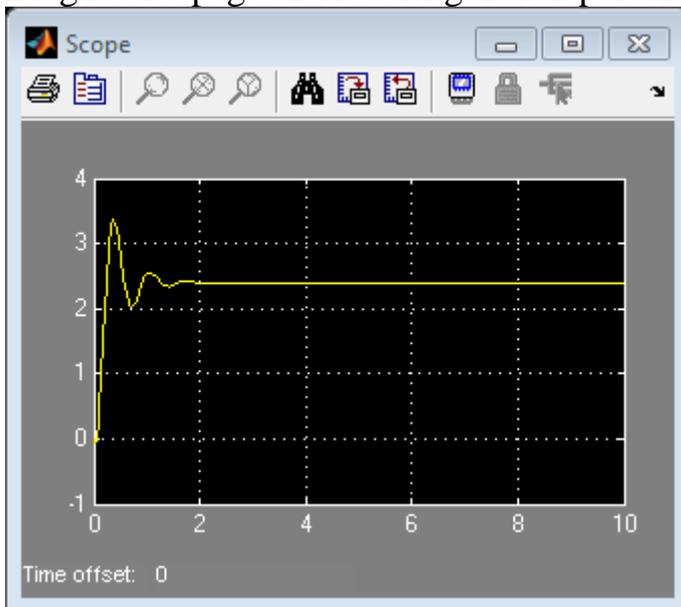
Transfer function:
 $0.02 s + 1$

 $0.003247 s^2 + 0.242 s + 1$

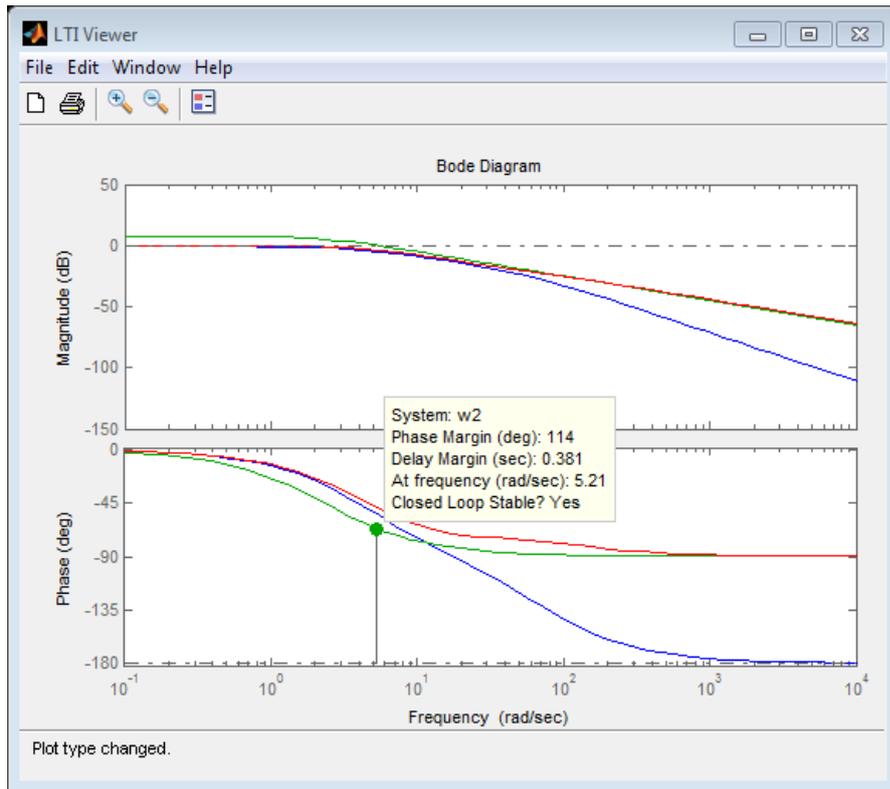
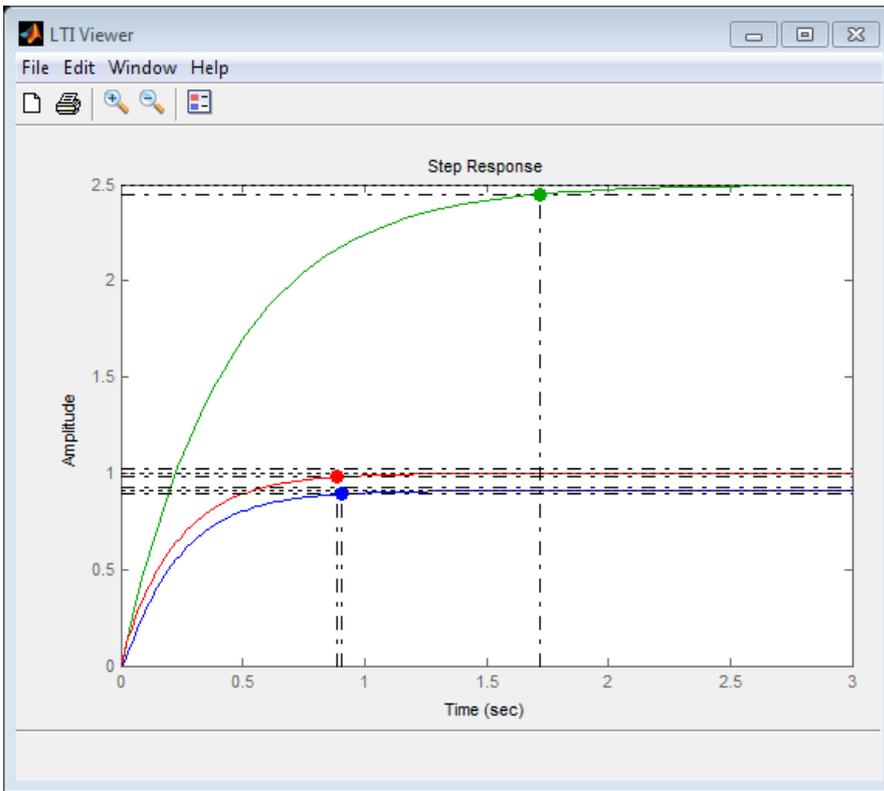
>>simulink



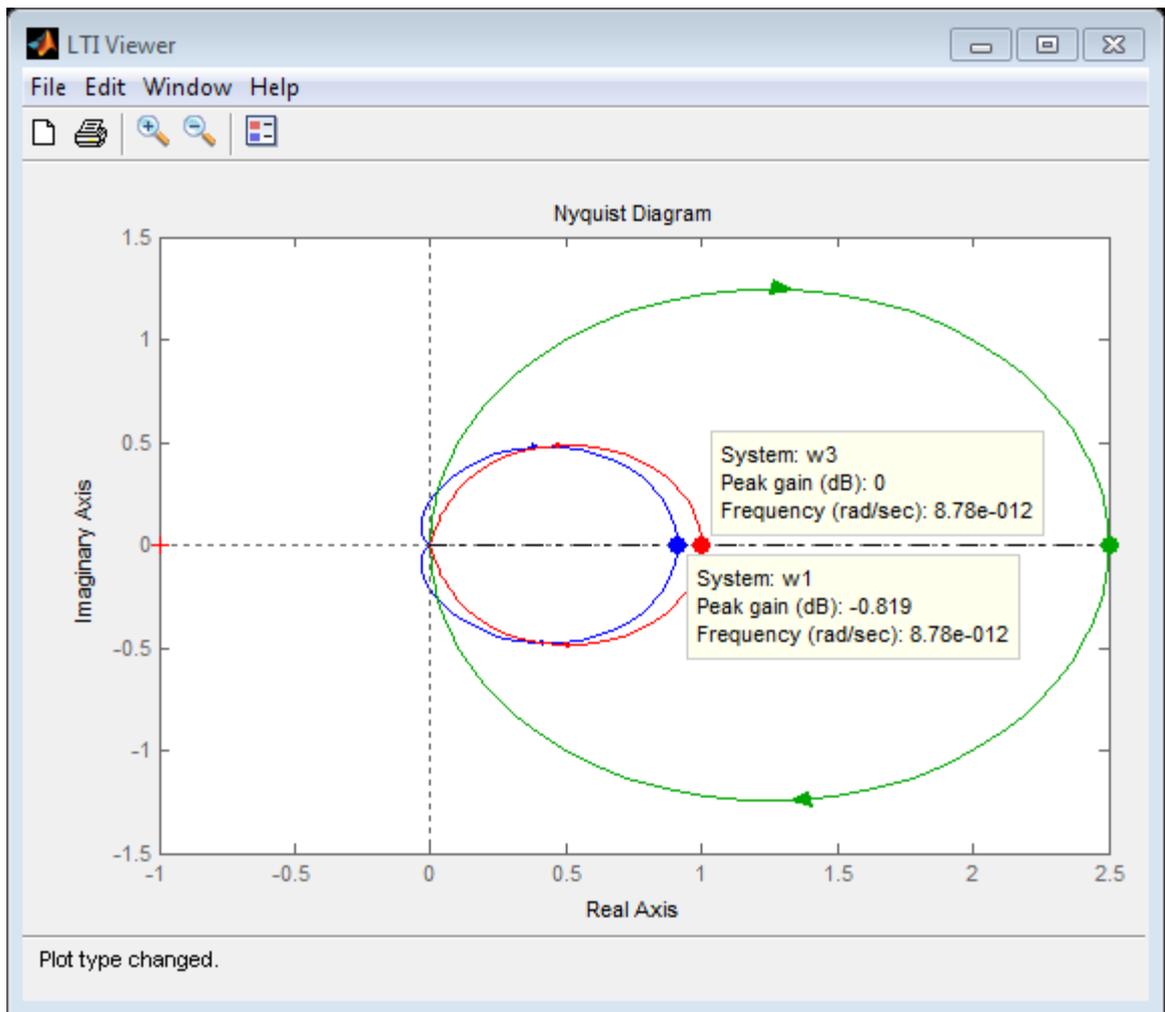
Warning: Using a default value of 0.2 for maximum step size. The simulation step size will be equal to or less than this value. You can disable this diagnostic by setting 'Automatic solver parameter selection' diagnostic to 'none' in the Diagnostics page of the configuration parameters dialog.



>>ltview



ЛАХ ва ЛФХ



ΑΦΧ

Список литературы

1. Методы классической и современной теории автоматического управления / Под. ред. К.А. Пупкова. Том 1-4 – М.: МГТУ им. Баумана. 2004 г.
2. Ротач В.Я. Теория автоматического управления. – М.: Изд-во МЭИ, 2004. -400 с.
3. Бесекерский В.А., Попов Е.П. Теория систем автоматического управления. СПб.: Профессия, 2004. -752 с.
4. Воронов А.А., Ким Д.П. Лохин В.М. и др. Теория автоматического управления. Учебник. 1,2 ч. – М.: Высш.шк., 1986.