#### Создание параметрической модели теплообменника

*Рассмотрен процесс разработки нового параметризованного блока-модели в теплогидравлической системе SimInTech, на основе базовых расчетных блоков и субмодели.*

Иногда в процессе создания теплогидравлических схем выясняется, что некоторые места на различных схемах идентичны по своему топологическому строению и различаются только, например, по геометрическим параметрам. Так же происходит и с оборудованием: например, различные модели подогревателей имеют одинаковый механизм работы (совпадающую расчетную, или нодализационную, схему), различаясь при этом только численными значениями параметров, таких как объем, высота трубчатки, высотная отметка и т.п. В таких случаях имеет смысл сделать предварительную модель, отладить ее, выяснить, какие свойства базовых блоков изменяются, а какие нет, а после этого создать на ее основе новый блок, который пользователи смогут применить при создании своих моделей. Используя такой блок, пользователю будет необходимо только поменять свойства этого блока для того, чтобы привести модель к соответствию тем параметрам, который должны быть. Так же пользователю облегчается задача контроля важных параметров базовых блоков, находящихся внутри, т.к. параметрам созданного блока могут быть присвоены параметры базовых блоков, и пользователю не обязательно проверять параметры базовых блоков при расчете.

Перед созданием нового блока на основе базовых блоков пользователь должен определиться с задачами, который должен решить данный блок и с тем, какой выходной результат должен получиться (т.е. вопросы, возникающие перед созданием нового теплогидравлического блока, абсолютно идентичны тем вопросам, которые встают перед пользователем во время создания нового блока в системе автоматики). После того как пользователь определил для себя ответы на данные вопросы, можно приступать к созданию нового блока.

Алгоритм создания блока таков:

1. Устанавливаем на схему блок «Субмодель МВТУ»;
2. Внутри субмодели устанавливаем нужное количество блоков «Порт входа TPP»/«Порт выхода TPP»;
3. Добавляем на схему базовые блоки из стандартной библиотеки TPP, расставляем их на схеме;
4. Соединяем блоки линиями связями;
5. Обновляем изображение блока;
6. Добавляем свойства и параметры блока;
7. Пишем скрипт для блока.
8. Отлаживаем блок автономно, например для номинального состояния на граничных условиях.

Рассмотрим в качестве примера создание модели теплообменника на основе базовых блоков.

Для этого:

1. Установим на схему блок «Субмодель МВТУ», сразу изменим тип элемента с «Субмодель МВТУ» на «Теплообменник TPP тип 1». Это ОБЯЗАТЕЛЬНАЯ процедура, которую необходимо проходить при создании нового блока. Меняем имя блока (см. **Рисунок 1**);
2. Заходим внутрь субмодели, устанавливаем там 2 блока «Входной порт TPP» и два блока «Выходной порт TPP» (см. **Рисунок 2**);
3. Изменяем имена блоков «Входной порт TPP» на «Вход: трубчатка» и «Вход: межтрубное пространство» (см. **Рисунок 3**). Для этого нужно выполнить двойной клик левой кнопкой мыши в поле блока, после чего появится окно, в котором изменяется имя (см. **Рисунок 4**);
4. Изменяем таким же способом имена блоков «Выходной порт TPP» на «Выход: трубчатка» и «Выход: межтрубное пространство» (см. **Рисунок 5**);
5. Ставим на схему два блока типа «Канал общего вида» (см. **Рисунок 6**);
6. Соединяем блоки «Входной порт TPP», «Выходной порт TPP» и «Канал общего вида» линиями связями (см. **Рисунок 7**);
7. У блоков «Канал общего вида» изменяем свойство «Тепловая связь» со значения «Нет» на «Да», после чего у каждого из блоков появится тепловой порт. Соединяем порты между собой (см. **Рисунок 8**);
8. Выходим из субмодели;
9. Вызываем меню субмодели одиночным кликом правой кнопки мыши по блоку, проходим на вкладку «**Общие**» → «**Графическое** **изображение**». В появившемся окне удаляем стандартное изображение и с помощью примитивов разрабатываем свое (**см. Рисунок 9**). В нашем случае изображение реализовано с помощью примитивов «Полилиния» и «Сектор». Сохраняем изображение и выходим из блока;
10. Добавляем нужные нам свойства (см. **Таблица 1**). Для этого выделяем блок и пройдем в главное меню «**Правка**» → «**Изменить** **блок**». Откроется окно редактирования свойств блока (см. **Рисунок 10**);
11. Вносим в категорию свойств для чтения некоторые свойства (данные свойства рассчитываются с помощью кода, написанного на внутреннем языке программирования): **F;S1;S2;Dg1;Dg2;V1;V2** (см. **Рисунок 11**);
12. Переходим на вкладку параметры, где вносим 8 параметров (см. **Таблица 2**);
13. Сохраняем внесенные свойства и параметры. После сохранения внесенные свойства и параметры можно увидеть в пунктах меню «Свойства объекта» и «Параметры объекта» (см. **Рисунок 12**);
14. Заходим внутрь субмодели на вкладку «Параметры», где прописываем следующий скрипт:

|  |  |
| --- | --- |
|  | //Установление тепловых связей в зависимости от типа течения  //(прямоток или противоток)  initialization  var stemp : string;  hc=1..Count;  if fType = 1 then hc=Count..1;  InterTube.Material=Material;  Tube.Material=Material;      Dg1 = d-2\*s;  if rType = 0 then Dg2=d\*(2\*sqrt(3)\*(sr/d)^2/pi - 1)  else Dg2=d\*(4\*(sr/d)^2/pi - 1);  F = pi\*(D-s)\*n\*L; //Поверхность теплообмена  S1= n\*Dg1\*Dg1\*pi/4; //Проходное сечение трубного пространства  S2=n\*Dg2\*D\*pi/4; //Проходное сечение межтрубья  V1=S1\*L; //Объем трубного пространства  V2=S2\*L; //Объем межтрубного пространства  setpropevalstring(submodel,"Dg1",Dg1);  setpropevalstring(submodel,"Dg2",Dg2);  setpropevalstring(submodel,"S1",S1);  setpropevalstring(submodel,"S2",S2);  setpropevalstring(submodel,"F",F);  setpropevalstring(submodel,"V1",V1);  setpropevalstring(submodel,"V2",V2);      setpropevalstring(Tube,"Count",Count);  setpropevalstring(Tube,"Gidr\_D","["+Count#Dg1+"]");  setpropevalstring(Tube,"Sechen","["+Count#(S1)+"]");  setpropevalstring(Tube,"Dlina","["+Count#(L/Count)+"]");  setpropevalstring(Tube,"Soprot","["+Count#0.0+"]");  setpropevalstring(Tube,"InvSopr","["+Count#0.0+"]");  stemp = "["; for (i=1,Count) begin  stemp = stemp + floattostr(Dz1/Count); if i < Count then stemp = stemp + ",";end; stemp = stemp + "]";  setpropevalstring(Tube,"Z",stemp);  setpropevalstring(Tube,"X","["+Count#0+"]");  setpropevalstring(Tube,"Y","["+Count#0+"]");  setpropevalstring(Tube,"Bor","["+Count#0+"]");  setpropevalstring(Tube,"Sten","["+Count#(s/2)+"]");  setpropevalstring(Tube,"F","["+Count#(F\*Kf1/Count)+"]");  setpropevalstring(Tube,"Rz1","["+Count#Rz1+"]");  setpropevalstring(Tube,"HeatElements","["+hc+"]");  setpropevalstring(InterTube,"Count",Count);  setpropevalstring(InterTube,"Gidr\_D","["+Count#Dg2+"]");  setpropevalstring(InterTube,"Sechen","["+Count#(S2)+"]");  setpropevalstring(InterTube,"Dlina","["+Count#(L/Count)+"]");  setpropevalstring(InterTube,"Soprot","["+Count#0.0+"]");  setpropevalstring(InterTube,"InvSopr","["+Count#0.0+"]");  stemp = "["; for (i=1,Count) begin  stemp = stemp + floattostr(Dz2/Count); if i < Count then stemp = stemp + ",";end; stemp = stemp + "]";  setpropevalstring(InterTube,"Z",stemp);  setpropevalstring(InterTube,"X","["+Count#0+"]");  setpropevalstring(InterTube,"Y","["+Count#0+"]");  setpropevalstring(InterTube,"Bor","["+Count#0+"]");  setpropevalstring(InterTube,"Sten","["+Count#(s/2)+"]");  setpropevalstring(InterTube,"F","["+Count#(F\*Kf2/Count)+"]");  setpropevalstring(InterTube,"Rz1","["+Count#Rz2+"]");  setpropevalstring(InterTube,"HeatElements","["+hc+"]");  InitObject(Tube);  InitObject(InterTube);    end;  Power1=abs(Tube.g\*(Tube.\_hvh-Tube.\_hvyh))\*4.182e-3;  dP1=abs(Tube.\_pvh-Tube.\_pvyh);  Tin1=Tube.\_tvh;  Tou1=Tube.\_tvyh;  Power2=abs(InterTube.g\*(InterTube.\_hvh-InterTube.\_hvyh))\*4.182e-3;  dP2=abs(InterTube.\_pvh-InterTube.\_pvyh);  Tin2=InterTube.\_tvh;  Tou2=InterTube.\_tvyh; |

После выполнения данных шагов создание блока «Теплообменник TPP тип 1» завершено.

При создании блока мы указали, что некоторые свойства блока должны быть только для чтения. Это было связано с тем, что свойства рассчитывались с помощью использования внутреннего языка программирования. Однако есть еще две категории, на которые можно поделить свойства. Обобщим и рассмотрим каждую категорию свойств.

Свойства можно разделить на следующие категории:

* видимые – эту категорию каждый пользователь может увидеть и задать при входе в меню свойств блока;
* невидимые – эта категория не видна из меню свойств блока. Задать значение данных свойств можно только в процессе редактирования свойств блока либо в случае, если пользователь знает имя этого свойства, с помощью использования скрипта;
* свойства для чтения – эта категория свойств видна из меню свойств блока, однако задать их нельзя. Данные свойства либо вычисляются в процессе расчета, либо задаются в процессе редактирования свойств блока, либо, если пользователь знает имя этого свойства, с помощью скрипта.

Использоваться в расчете могут все три категории свойств.

Для изменения категории свойств, нужно зайти в меню редактирования свойств блока, где в строках «Невидимые свойства» и «Свойства для чтения» перечислить имена тех свойств, которые должны быть отнесены к той или иной категории (см. **Рисунок 13**).