

УДК 658.512, 004.942

АВТОМАТИЗИРОВАННОЕ ПРОЕКТИРОВАНИЕ ЭЛЕКТРОННЫХ МАКЕТОВ ЭЛЕМЕНТОВ СБОРОЧНОЙ ОСНАСТКИ ПОСРЕДСТВАМ ПРОГРАММНОГО МОДУЛЯ NX/OPEN API

© 2013 М.В. Лаврентьева, Х.В. Чьен

Национальный исследовательский
Иркутский государственный технический университет

Поступила в редакцию 15.11.2013

Широкое распространение CAD/CAE систем при решении большинства конструкторских и технологических задач дает возможность для развития систем автоматизированных проектирования. В данной статье описываются два метода автоматизации проектирования сборочной оснастки с использованием средств NX: Siemens PLM Software, NX/Open API. Применение подобной системы позволяет оптимизировать трудовые затраты инженера.

Ключевые слова: *автоматизация, сборочная оснастка, CAD/CAE системы, технологическая подготовка производства, NX/Open API*

Технологической подготовке сборки самолета всегда уделялось наибольшее внимание ввиду ряда факторов:

- окончательная точность самолета напрямую зависит от качества выполнения сборочных процессов;
- наличие большого числа и разнообразие типов сборочных приспособлений;
- небольшой процент автоматизации при сборке планера самолета и технологической подготовке сборочного производства;

Разработка системы автоматизированного проектирования (САПР) сборочных приспособлений позволяет сократить время, затрачиваемое инженером-конструктором на разработку сборочной оснастки в разы, так как практически вся проектная часть будет записана в программном коде и станет требовать от пользователя лишь указания базовых плоскостей, осей, а также габаритных размеров каждого элемента сборочного приспособления. Автоматизация проектно-конструкторских работ является процессом достаточно сложным, так как требует владения не только инженерными знаниями, но и навыка программирования, притом программирования не в чистом виде, а с использованием средств NX: Siemens PLM Software, точнее один из его программных модулей NX/Open API (Application Program Interface), имеющий в своем арсенале Program Interface), имеющий в своем арсенале

- доступ к объектам геометрической модели для программных приложений разработчиков или программ отдельных пользователей;
- возможность программным способом создавать геометрические модели, получать информацию об объектах, формировать сборки, генерировать чертежную документацию и т.д.

Основная цель разработки САПР заключается в создании программного продукта, способного самостоятельно создавать электронный макет сборочного приспособления, в идеале без участия, но под контролем конструктора. В данной работе будут рассмотрены два метода автоматизированного проектирования на примере простой прямоугольной рамы и нестандартной опоры сборочного приспособления.

Первый метод представляет собой процесс, полностью повторяющий традиционное проектирование рамы, написанный на языке C++. Рама сборочного приспособления относится к группе несущих элементов сборочного приспособления (СП), образующих его каркас. Как правило, рама состоит из балок, каждая балка представляет два сваренных между собой швеллера либо трубу квадратного сечения. Для начала работы необходимо выделить элементы КЭМ, используемые для создания рамы, которые должны в обязательном порядке иметь файл сборки: плоскость симметрии рамы (datum plane); горизонтальная плоскость рамы (datum plane); поперечная плоскость рамы (datum plane); продольная ось рамы (datum axis); вертикальная ось рамы (datum axis); поперечная ось рамы (datum

Лаврентьева Мария Вячеславовна, аспирантка. E-mail: mira.amazon@gmail.com
Ха Ван Чьен, аспирант. E-mail: maimaibenet_0113@yahoo.com

axis). Наличие данных базовых элементов, имитирующих систему координат самолета, обеспечивает необходимое положение рамы в пространстве. Все перерасчеты координат система должна производить относительно этой СК. Таким образом, выявив все необходимые составляющие, можно приступить к построению.

Первоначально система строит эскиз, используя лишь габаритные размеры рамы, вводимые пользователем. Каждая линия такого эскиза будет являться осью симметрии сечения отдельной балки (рис. 1а и 1б).

```

/** Создание эскиза */
theSession->Preferences()->Sketch()->SetCreateInferredConstraints(true);
theSession->Preferences()->Sketch()->SetContinuousAutoDimensioning(false);
theSession->Preferences()->Sketch()->SetDimensionLabel(Preferences::SketchPreferences::DimensionLabelTypeExpression);
theSession->Preferences()->Sketch()->SetTextSizeFixed(true);
theSession->Preferences()->Sketch()->SetFixedTextSize(4.0);
theSession->Preferences()->Sketch()->SetDisplayObjectColor(false);

Sketch *nullSketch(NULL);
SketchInPlaceBuilder *sketchInPlaceBuilder;
sketchInPlaceBuilder = workPart->Sketches()->CreateNewSketchInPlaceBuilder(nullSketch);
sketchInPlaceBuilder->SetCreateIntermediateDatumCsys(false);
sketchInPlaceBuilder->SetMakeOriginAssociative(false);

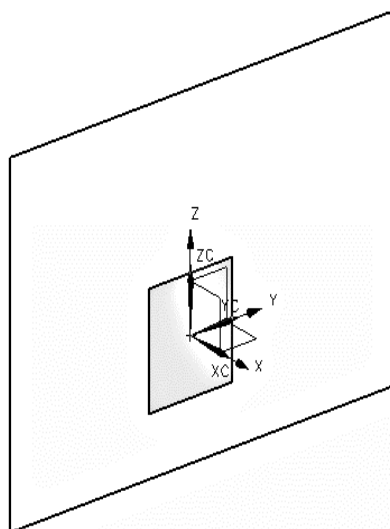
DatumPlane *datumPlane = framePlane;

sketchInPlaceBuilder->PlaneOrFace()->SetValue(datumPlane);
sketchInPlaceBuilder->Axis()->SetValue(dynamic_cast<DatumAxis *>(bases[3]));
sketchInPlaceBuilder->SetReverseAxis(false);

sketch=(dynamic_cast<Sketch *>(sketchInPlaceBuilder->Commit()));

Features::Feature *sketchFeature;
sketchFeature = sketch->Feature();
sketch->Activate(Sketch::ViewReorientFalse);
sketchInPlaceBuilder->Destroy();
    
```

а)



б)

Рис. 1. Процесс создания эскиза сборочного приспособления:

а – программный код построения эскиза, б – эскиз рамы сборочного приспособления

По существующему эскизу расставляются эскизы сечения балки, созданные аналогично эскизу рамы, все размеры сечений заложены в базе данных, что позволяет системе самой выбирать нужное нужной типоразмер балки, отталкиваясь от заданных конструктором значений. По окончании осуществляется операция «вытягивания», эскизы сечения протягиваются по эскизу рамы (рис. 2, 3).

Второй метод приведен на примере нестандартной опоры, не имеющей размерного ряда, который мог быть заложены в базу данных. Конструкция опоры остается неизменной, в связи с чем для ее программного написания наиболее рациональным решением будет применение метод «клонирования». Это позволяет,

используя одну разработанную в соответствии со всеми предъявляемыми предприятием требованиями модель (рис. 4), копировать ее в рабочую папку конструктора с заданными им габаритными размерами. Все зависимости (связи) между элементами опоры задаются в NX рис. 5, а сам процесс клонирования записан в программном коде (рис. 6). Такой подход является универсальным, поскольку позволяет работать с любыми элементами конструкции при условии, что все возможные связи будут заданы непосредственно в физическом файле NX. Однако таким образом можно работать лишь с типовыми моделями, чья конструкция не требует изменения или доработки.

```

8:ASSY:FrameBuilder
CreateSketch()
std::vector<NXObject*>Refs(3);
std::vector<int>ReverseDir; // по сути для квадратной трубы определяет с какой стороны от линии будет построения (внутри
switch (frame->frame_type) // тип рамы, швеллер или квадрат
{
case 0:
    Refs[0]=sketchLine[0];
    Refs[1]=sketchLine[0];
    Refs[2]=this->bases[5];
    Refs[3]=this->bases[2];
    ReverseDir.resize(1);
    ReverseDir[0]=2;
    mCreateProfile(2,Refs,ReverseDir,offset_bufM,"0");
    ReverseDir[0]=1;
    mCreateProfile(2,Refs,ReverseDir,offset_bufM,"0");

    Refs[0]=sketchLine[1];
    Refs[1]=sketchLine[1];
    Refs[2]=this->bases[5];
    Refs[3]=this->bases[2];
    ReverseDir.resize(1);
    ReverseDir[0]=2;
    mCreateProfile(2,Refs,ReverseDir,offset_bufM,offsetV);
    ReverseDir[0]=1;
    mCreateProfile(2,Refs,ReverseDir,offset_bufM,offsetV);

    Refs[0]=sketchLine[2];
    Refs[1]=sketchLine[2];
    Refs[2]=this->bases[5];
    Refs[3]=this->bases[2];
    ReverseDir.resize(1);
    ReverseDir[0]=2;
    mCreateProfile(2,Refs,ReverseDir,offset_bufM,offsetV);
    ReverseDir[0]=1;
}
    
```

Рис. 2. Построение рамы сборочного приспособления

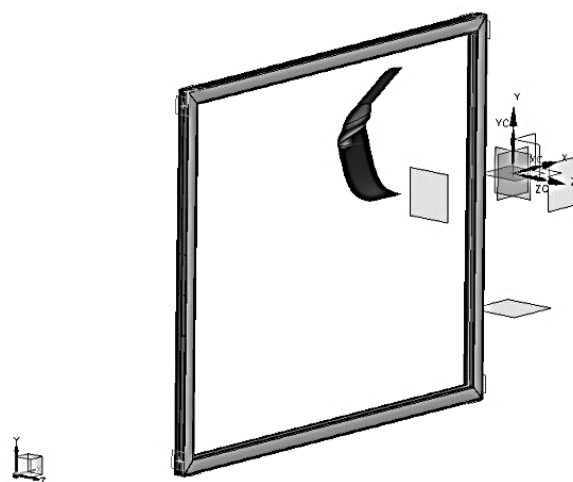


Рис. 3. Рама сборочного приспособления

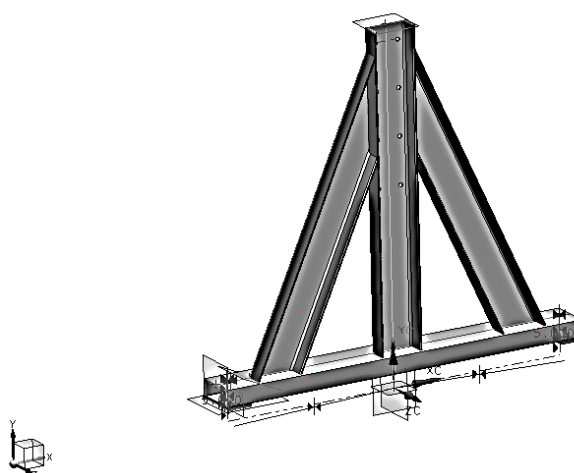


Рис. 4. Модель нестандартной опоры

Имя	Формула	Знач...	Един...	Тип	Коммента...	Проверки
b (SKETCH_000:Эскиз(3) Горизонтальный разм...	b_otv+2*b_otst	2600	мм	Число		
p614 (SKETCH_003:Эскиз(35) Горизонтальный ...	b_pr	180	мм	Число		
p701 (SKETCH_005:Эскиз(57) Горизонтальный ...	b_pr	180	мм	Число		
p709 (SKETCH_006:Эскиз(59) Горизонтальный ...	b_pr	180	мм	Число		
a1 (Простое отверстие(25) Вертикальный раз...	b_pr/2	90	мм	Число		
b1 (SKETCH_000:Эскиз(3) Горизонтальный раз...	b_pr/2	90	мм	Число		
p546 (Вытягивание(28) End Limit)	b_pr/2+10	100	мм	Число		
D_gor (Простое отверстие(61) Diameter)	d	17	мм	Число		
a (SKETCH_000:Эскиз(3) Вертикальный размер ...	h-h-120	-120	мм	Число		
h1 (SKETCH_000:Эскиз(3) Вертикальный разме...	h-h-500	-500	мм	Число		
h_pr (SKETCH_000:Эскиз(3) Перпендикулярны...	if(b_pr==50)32 ...	70	мм	Число		

Рис. 5. Задание связей в системе NX

```

void ST0318::CloneBuilder::Clone(Product *product, const char *assy)
{
    UF_initialize();
    char assyName[255+1];
    UF_CLONE_set_def_action(UF_CLONE_clone);
    UF_CLONE_set_def_directory(NXApplication::projectData->GetProjectDir().c_str());
    UF_CLONE_set_def_naming(UF_CLONE_user_name);
    UF_CLONE_set_family_treatment(UF_CLONE_treat_as_lost);
    UF_CALL(UF_CLONE_initialize( /*UF_CLONE_edit_operation*/UF_CLONE_clone_operation));

    UF_PART_load_status_t load_status;
    UF_CALL(UF_CLONE_add_assembly(assy, &load_status));
    if(load_status.failed)
    {
        string fileName=this->component->GetProduct()->GetPrototypePartName().c_str();
        for(int i=0; i<load_status.n_parts; ++i)
        {
            if(fileName == string(load_status.file_names[i]))
                return;
        }
    }
    char *oldName;
    UF_CLONE_part_state_t state;
    std::vector<string>Complete;
    UF_CLONE_start_iteration();
    do
    {
        UF_CALL(UF_CLONE_iterate(&oldName));
        if(oldName)
        {
            UF_CLONE_ask_part_state(oldName, &state);
            if(state==UF_CLONE_present_state)
            {
                Complete.push_back(oldName);
            }
        }
        else
    }
}
    
```

Рис. 6. Реализация метода клонирования

Для того чтобы обеспечить изменение пространственного положения, как рамы, так и опоры, задаются данные о матрицах расположения каждого элемента СП в формате 4x4, что дает возможность базировать один элемент относительно другого. Таким образом, система, в коде опоры имея ссылку на матрицу рамы (рис. 7), перемножает ее с матрицей опоры, получая в результате матрицу со значениями, определяющими конечное положение опоры относительно рамы с соответствующим знаком.

Выводы: автоматизация процесса проектирования не только ускорит работу конструктора, но и унифицировать процесс проектирования сборочной оснастки. Помимо этого, все расчеты и построение будут проводиться машиной, что исключает возможность ошибок из-за так называемого человеческого фактора.

```

/** Рама */
Frame* Frame=new Frame();
Frame->CommonInit();
Component* componentFrame=new Component(NULL, Frame);
Matrix4d matrix;
matrix<<0,0,1,0,
        1,0,0,0,
        0,1,0,0,
        50,100,300,1;
componentFrame->SetLocation(matrix);
    
```

Рис. 7. Определение пространственного положения рамы

Представленная в рамках данной статьи работа проводится при финансовой поддержке правительства Российской Федерации (Минобрнауки России) по комплексному проекту 2012-218-03-120 «Автоматизация и повышение эффективности процессов изготовления и подготовки производства изделий авиатехники нового поколения на базе Научно-производственной корпорации «Иркут» с научным сопровождением Иркутского государственного

технического университета» согласно постановлению Правительства Российской Федерации от 9 апреля 2010 г. №218.

СПИСОК ЛИТЕРАТУРЫ:

1. Ахатов, Р.Х. Выбор опорных базовых точек при определении схемы базирования сборочной единицы / Р.Х. Ахатов, Х.В. Чьен // Вестник МАИ. 2013. Том 20. №3. С. 110-118.
2. Григорьев, В.П. Приспособления для сборки узлов и агрегатов самолетов и вертолетов / В.П. Григорьев, Ш.Ф. Ганиханов – М.: Машиностроение, 1977. 138 с.
3. Камаев, В.А. Технология программирования / В.А. Камаев, В.В. Костерин – М.: Высшая школа, 2005. 360 с.
4. Краснов, М. Unigraphics для профессионалов / М. Краснов, Ю. Чигишев – М.: Издательство «Лори», 2004. 274 с.
5. Официальный сайт компании Siemens PLM Software (UGS) [Электронный ресурс] – Режим доступа: <http://www.plm.automation.siemens.com>.

AUTOMATED ELECTRONIC MODELS OF ELEMENTS OF ASSEMBLY EQUIPMENT BY MEANS OF THE PROGRAM NX/OPEN API MODULE

© 2013 M.V. Lavrentyeva, H.V. Chyen

National Research Irkutsk State Technical University

The wide spread occurrence of CAD/CAE systems at solution the majority of design and technological tasks gives the chance for development the systems of automated design. In this article two methods of automation the design of assembly equipment with use of NX means are described: Siemens PLM Software, NX/Open API. Use of similar system allows to optimize the labor costs of engineer.

Key words: *automation, assembly equipment, CAD/CAE system, technological preparation of production, NX/Open API*

Лаврентьева Мария Вячеславовна, аспирантка.

E-mail: mira.amazon@gmail.com

Ха Ван Чьен, аспирант.

E-mail: maimaibenem_0113@yahoo.com