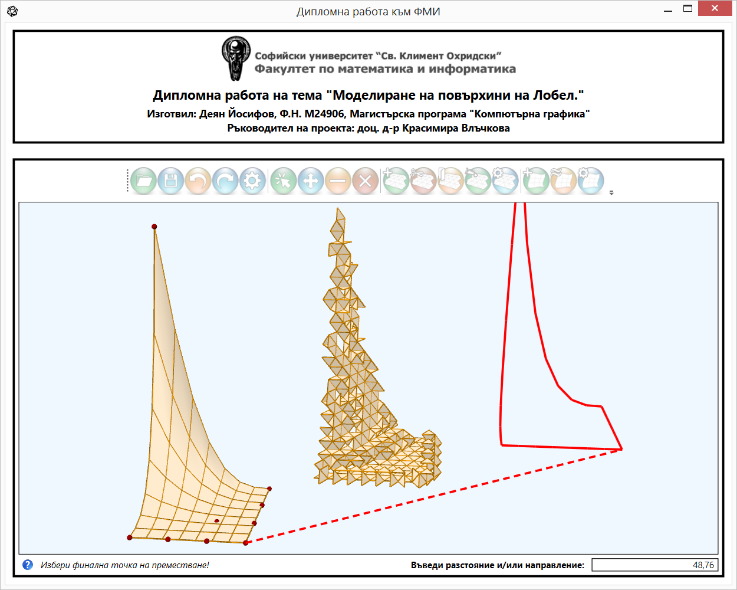


**Моделиране на повърхнини на Лобел**

****

**Cъдържание**

1. [Увод](#_Увод).
   1. [Проблеми на „Не-Евклидовата“ архитектура](#_Проблеми_на_„Не-Евклидовата).
   2. [Повърхнини на Лобел](#_Повърхнини_на_Лобел).
   3. [Описание на поставените проблеми](#_Описание_на_поставените).
2. [Математически метод и алгоритми](#_Математически_метод_и).
3. Структури от данни използвани при имплементацията.
4. Напътствия при използване на приложението.
5. Библиография.

# Увод

С напредването на технологиите все по-често забелязваме модерни архитектурни явление характерни с техните разчупени форми, различаващи от стандартния за архитектурата паралелепипед. Сред известните имена на архитекти търсещи нестандарните решения са Франк Гери и Заха Хадид, но и мнозина други архитекти от нашето съвремие. Често пъти, коректно или не, този тип архитектура бива наречен „Не-Евклидова архитектура“ заради кривите форми на използваните повърхнини. Примери на някои такива проектни решения могат да бъдат видяни на следващите картинки.



## Проблеми на „Не-Евклидовата архитектура“

Проектирането и последващото практическо изпълнение на този тип архитектура в повечето случаи струва в пъти повече от стандартната архитектура. Оскъпяването идва от няколко различни фактора:

* **Сложности при самото проектиране**. Моделирането на такива форми изисква по-сложни изчисления касаещи отделните елементи и съгласуването на инсталации от различни специалности. Отделно изработването на детайли и четими монтажни схеми е нелека задача, тъй като често пъти всеки отделен елемент и детайл е различен от другите.
* **Сложност при изработката на материалите**. Ако формата не бъде разчленена на някакви равнинни парчета, това често пъти прави изработката на отделните елементи доста трудна и скъпа. Дори и отделните парчета да са равнинни, наличието много различни видове форми на всяка от частите също оскъпява и забавя допълнително производствения процес.
* **Сложност при строежа на архитектурния обект**. Сглабянето на всичките различни елементи на сградата може да се окаже доста сложна задача, дори при наличието на добри монтажни чертежи и схеми. Това изисква високо квалифицирана работна ръка, която е по-скъпа. Отделно самият процес е по-бавен от конвенционалното строителство, а както знаем „времето е пари“.

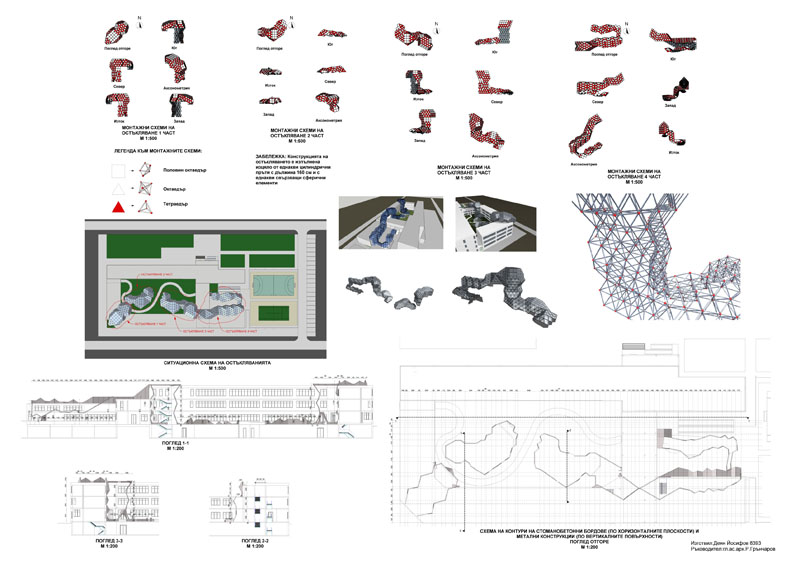
Всички тези оскъпяващи проблеми биха могли да се решат ако сложната структура се изгради от максимален брой еднакви елементи с минимален брой разлики в монтажните детайли между отделните парчета. Едно такова решение са така наречените повърхнини на Лобел, за които следва да обясним какво представляват.

## Повърхнини на Лобел

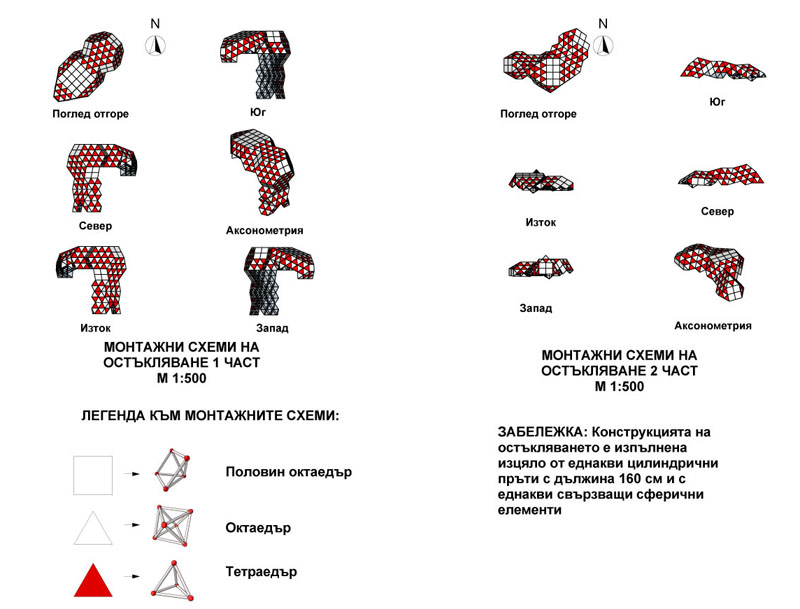
В близките няколко години френският архитект Ален Лобен работи по изследването на архитектурни структури изградени изцяло от еднакви равностранни триъгълници. Вероятно други хора също са се сблъсквали с този въпрос, но разработките на този френски архитект в неговия web site <http://www.equilatere.net/> постепенно налагат името на такива структури като „Повърхнини на Лобел“ или на английски – „Lobel Frames”.

Самият аз като студент по архитектура през 2011 година разработих преддипломен работен проект на реконструкция на НПМГ „акад. Любомир Чакалов“, който проект използваше метална конструкция изградена изцяло от еднакви равностранни триъгълници. По това не бях чувал за разработките на Ален Лобел, макар и да бях търсил доста в интернет за проучване разработки на подобни конструкции. Това ме води на мисълта, че вероятно разработките на френския колега са сравнително скорошни и дори да са били налични през 2011, то те са били твърде нови в интернет пространството, за да бъдат откриваеми.

По-долу може да видите едно от таблата към моя проект, което има ситуация, изгледи, перспективи и монтажни схеми на въпросната конструкция.



Особено внимание бих искал да обърна с по-близък поглед към част от монтажните схеми, които разясняват простотата на въпросната конструкция.



Както може да се види от легендата цялата конструкция е изградена от еднакви октаедри и тетраедри, които в крайна сметка оформят една повърхнина от еднакви равностранни триъгълници. Тази повърхнина позволява да бъде изградена изцяло от еднакви пръти свързани в еднакви сверични възли. Тази еднаквост на елементите позволява едновременно бързо производство, бърз и лесен монтаж на мястото на обекта следвайки простите двуцветни монтажни схеми (червен триъгълник означава тетраедър, а бях означава октаедър, както е видно и от легендата).

Единствената трудност, която остана нерешена е сложността на проектирането на въпросната конструкция. По времето, което аз работих този проект нямаше софтуер, който да ми помага с лесното моделиране на такъв тип конструкция. Това, което направих тогава беше да напиша един просто plug-in за Google Sketch Up, който да може от наличен равностранен триъгълник да вдига от него тетраедър, октаедър или икосаедър. Този плъгин може да бъде видян в моя Github акаунт: <https://github.com/deyan-yosifov/Deyan-Projects/tree/master/Ruby/SketchUpPlugins/dpy-equilateral>. Макар и с този плъгин, моделирането на цялостната структура беше доста трудоемка задача. Първоначално измоделирах една крива произволна повърхнина с известна програма, така че формата да отговаря на практическите и визуални цели, които търсех. След това вкарах модела в Google Sketch Up и започнах да добавям и трия тетраедърчета и октаедърчета едно по едно, така че получената структура да доближава първоначално измоделираната повърхнина. Добавянето и триенето на триъгълници един по един беше наистина времеемко, но тогавашните ми слаби познания по програмиране не ми позволиха да автоматизирам този процес.

И тук идва темата на сегашната дипломна работа, целяща да се улесни процеса при моделирането и проектирането на такива структури от еднакви триъгълници.

## Описание на поставените проблеми

Основната цел на сегашната дипломна работа е разработка на софтуер, който да улесни проектирането и моделирането на повърхнини на Лобел. Този софтуер ще предоставя следните възможности:

* **Моделиране на повърхнина на Лобел от една равнинна плоскост с равностранни триъгълници на принципа на известното японско изкусктво „киригами“.** На кратко обяснено киригами представлява рязане, огъване и лепене на лист хартия, така че да се измоделира някаква по-сложна триизмерна структура. На практика всяка повърхнина на Лобел може да се измоделира от един достатъчно голям лист хартия, който е разграфен с мрежа от еднакви равностранни триъгълници. Наличието на операции „рязане“, „огъване“ и „лепене“ на такива разграфени равнинни елементи, би позволило на ползвателя на софтуера да моделира или модифицира дадени повърхнини на Лобел.
* **Предоставяне на алгоритми за приближаване на дадена произволна повърхнина с повърнина на Лобел.** Докато моделирането със способите на киригами може да се окажа трудоемка и времеемка задача, то предлагането на такива алгоритми би позволо да се измоделира първо една произволна повърхнина и след това с един клик алгоритъмът да предложи приближение на въпросната повърхнина с мрежа от еднакви равностранни триъгълници.
* **Предоставяне на операции за отваряне и записване на работата в определени тримерни файлови формати.** Това е ключово поради три причини. Първо – за да може да моделира обекти от реален проект, ползвателя на софтуера трябва да може да вкара в програмата своя първоначален проект, в който иска да интегрира повърхнината на Лобел. Второ – когато човек използва софтуера би следвало да има възможност да запише работата си, за да може в последствие да я продължи. Трето – когато работа по моделирането е завършена, тя би следвало да може да бъде записана в известен файлов формат, който да може да бъде отварян от други програми, където работата по моделирането и интегрирането на геометрията може да продължи.
* **Предоставяне на операции за връщане или повтаряне на определена стъпка от работа с програмата.**Тези операции в други програми са известки като Undo-Redo и за ключови от гледна точка, че грешна стъпка по време на моделирането би трябвало да може да бъде лесно върната, а не да трябва да се започва работата от начало.

# Математически метод и алгоритми

В тази секция ще опиша идеите, реализирани в текущата имплементация на приложението, както и математическите изчисления необходими за осъществяването на алгоритмите.