

Использование параллельных вычислений при трехмерном конструировании лекал одежды

Введение

Конструирование лекал одежды для промышленного поточного производства и для индивидуальных заказов является достаточно специфической задачей. Обучение конструкторов длительный и трудоемкий процесс, и не все обучаемые становятся профессиональными конструкторами. Использование компьютеров должно помочь профессионалам раскрыть свои способности.

Расчетные методы проектирования лекал одежды существуют, вероятно, с 17 века. И до настоящего времени, не смотря на наличие различных методик и расчетных методов, разработка лекал одежды является трудоемким процессом, требующим большого опыта работы. Существующие плоскостные методики разработки лекал практически себя исчерпали, т.е. дальнейшее их совершенствование не позволяет получать гарантированный результат для всех типов фигур и всех покровов. Вместе с тем требуется постоянная разработка новых лекал в соответствии с изменением моды и антропологии населения. Оценка работы конструктора всегда субъективна и неоднозначна.

Для разработки лекал используются САПР, которые можно разделить на несколько условных типов:

традиционные, используемые только в промышленном производстве (Лектра, Гербер, Инвентроника, Тукатех, Ассист, отечественные Комтенс, Ассоль, украинские САПРЛЕГПРОМ, Грация и т.д.) использующие лекала базового размера разработанные вручную, вводимые в компьютер при помощи дигитайзера, автоматизирующие размножение лекал по размеро-ростам и раскладку лекал (проектирование настила);

параметрические (ЛЕКО, Грация, Графис и т.д.) использующие для описания лекал формулы и «язык программирования» для конструкторов.

В настоящее время параметризация включается во многие традиционные САПРы. Но при этом и ручная и автоматизированная разработка лекал остается плоскостной. Сложность использования трехмерных моделей для разработки связана с отсутствием теории и практических результатов в этой области. Результат работы конструктора – набор плоских лекал, а оценка работы конструктора производится на готовом отшитом изделии (неплоская оболочка) которое одевается на трехмерный манекен или человека, и оценивается как в статике, так и в динамике. Существуют системы с упрощенными методиками и фрагментарным использованием трехмерной информации, пригодные только для отдельных элементов (ЛЕКО, СТАПРИМ, Ассоль). В полной постановке задача конструирования даже не рассматривается.

Легкая промышленность является одной из наиболее динамичных отраслей, большое значение в которой играет скорость разработки. По некоторым изделиям «проектирование - производство – продажа» время исчисляется днями, при актуальности производства 2-3 месяца («попадание в сезон»).

Для ускорения разработки и повышения качества создания лекал одежды предлагается использовать развертку трехмерной модели одежды, спроектированной на основе трехмерной модели человека или опорной поверхности. Развертка получается численным методом путем решения краевой задачи с набором дополнительных ограничений и свойств ткани.

В полной постановке задача проектирования одежды может быть решена на компьютере, который в настоящее время относится к классу суперкомпьютер. Однако прогресс в области микроэлектроники в скором времени переведет компьютеры с такой производительностью в разряд доступных для малых предприятий и индивидуальных пользователей.

В настоящее время трехмерные модели одежды используются в основном не для разработки лекал, а при визуализации готовых моделей. При внешнем эффекте такая визуализация дает достаточно мало информации пользователю, и (оценочно) не дает большой коммерческой отдачи ни покупателю, ни продавцу САПР. При визуализации используют в основном простые однослойные модели, моделируется ткань средней группы растяжимости.

Визуализация – процесс (оценочно) более быстрый чем развертка, занимает (оценочно) 2-5 мин на быстром компьютере (системный блок до 3 тыс.дол), при величине сетки аппроксимации лекал около 1-2 см. Для точного моделирования поведения ткани необходима величина шага аппроксимации на порядок меньше – 1-2 мм. При этом объем вычислений и объем данных возрастает в 100 раз, т.е. время счета значительно превысит 200-500 мин (3-9 часов). При тенденции увеличения скорости компьютеров в 2 раза за 1.5 года приемлемое время расчета (1-2 мин) будет достигнуто через 6-7 лет. Используя параллельные вычисления можно сократить время расчета, но при этом возрастает сложность программирования связанная с распараллеливанием задачи.

Перечисленные проблемы являются общими для решения задач численными методами, и такие оценки вероятно в настоящее время останавливают от решения задачи развертки лекал «в лоб». Для решения данной задачи «в лоб» в приемлемое для конструктора время требуется производительность компьютера (оценочно) 2-3 тфлпс (небольшой суперкомпьютер), при этом результат не будет иметь экономического эффекта, т.е. себестоимость работ с использованием такой техники будет гораздо выше стоимости результата. Для примера, лаборант изготавливает макет швейного изделия за 2-3 часа (при малой автоматизации время сокращается до 20-30 минут). Качество такой реальной «визуализации» значительно выше существующих компьютерных моделей.

Однако, используя дополнительную информацию о задаче и знание предметной области можно значительно сократить объем вычислений. Использование эффективного оптимизирующего компилятора для многоядерных процессоров, распараллеливание задачи позволит выйти на время расчета 1-2 мин, необходимое для работы промышленных конструкторов и частных пользователей (системный блок до 3 тыс.долларов, двухпроцессорный многоядерный, с быстрой памятью).

Аналогичные предположения могут быть и у любого другого разработчика. Поэтому появление первых промышленных решений «разверток трехмерных лекал одежды» это вопрос времени. И первые эффективные реализации «разверток лекал» имеют шанс на коммерческий успех.

Система трехмерного конструирования лекал одежды

Описываемая ниже система трехмерного конструирования лекал одежды создавалась на базе плоскостной версии системы проектирования лекал ЛЕКО фирмы «Вилар софт». Система ЛЕКО была разработана на языке программирования Паскаль. В 2007 году программа переведена на язык Фортран

и была проведена оптимизация скорости выполнения программы. За выдающиеся успехи в разработке и оптимизации приложений для новейших платформ Intel в 2008 году фирма «Вилар софт» была награждена в рамках программы Intel Software Partner.

Изначально весь проект был нацелен на высокопроизводительные вычисления с использованием многоядерных/многопроцессорных систем. Для этого был выбран язык программирования наилучшим образом поддерживающий параллельные вычисления – Фортран (стандарт 95, 2003), выбран оптимизирующий компилятор наилучшим образом поддерживающий современные процессоры – компилятор фирмы Интел, осуществлен переход с языка Object Pascal на Intel(R) Visual Fortran Compiler. Использование компилятора Intel(R) Visual Fortran Compiler Professional Edition 11.1 позволяет в полной мере использовать все возможности современных процессоров, системы команд и расширения до SSE4.2. Перевод внутренних структур в векторный вид, использование массивов (с эффективной обработкой их Фортраном), позволили в полной мере использовать все возможности современных процессоров. Анализ программы с использованием VTune позволил выделить и повысить скорость выполнения критичных участков программы. С точки зрения реализации, это практически единственная коммерческая система для широкого рынка, использующая компилятор Фортрана фирмы Intel (по нашей информации).

Отсутствие каких либо аналогов и просто наработок в области трехмерной разработки одежды требует дополнительной исследовательской работы. Работа над системой ведется по 3 направлениям: разработка алгоритмов проектирования для САПР, создание и оптимизация программы, разработка алгоритмов проектирования лекал.

Система ЛЕКО, если кратко, состоит из базы данных размерных признаков, интерпретатора языка конструирования, редактор+просмотр результатов 2D-3D построений, базы данных лекал и системы проектирования раскладок лекал (настилов).

С точки зрения программирования задача выглядит как интерпретатор алгоритма, записанного на специальном языке программирования для работы с графическими объектами. Конструктор вносит изменения в интерпретируемый алгоритм-программу и при нажатии на клавишу исполнения должен максимально быстро увидеть результат на экране. Практически все операторы такой программы выполняются быстро, что позволяет получить время реакции для построения двумерных-трехмерных линий и контуры лекал при плоскостном конструировании за доли секунды. Оптимизированные алгоритмы генераторов плоских сеток так же позволяют получить заливки контуров на современных компьютерах за доли секунды (триангуляция замкнутых контуров произвольной формы с достаточно малым шагом сетки, отображение полупрозрачных лекал с использованием OpenGL). Сложности возникают при работе с трехмерными поверхностями – генерация трехмерной сетки из пространственной границы сложной формы, формирование трехмерной поверхности с заданными граничными условиями, введение параметризации, для возможности создания параметрической поверхности с нужными свойствами, оцениваемыми преимущественно визуально – все это требует большого объема вычислений. Далее следует объединение-разбиение сформированных поверхностей и развертка – получение плоских контуров лекал. Трехмерные поверхности в основном не разворачиваемы, поэтому развертка является неким приближением с заданными свойствами. Следующий этап – визуализация плоских контуров в пространстве.

Наиболее ресурсоемкими являются операторы формирования трехмерной

сетки, развертка и визуализация. Все они реализованы численными методами итерационными алгоритмами.

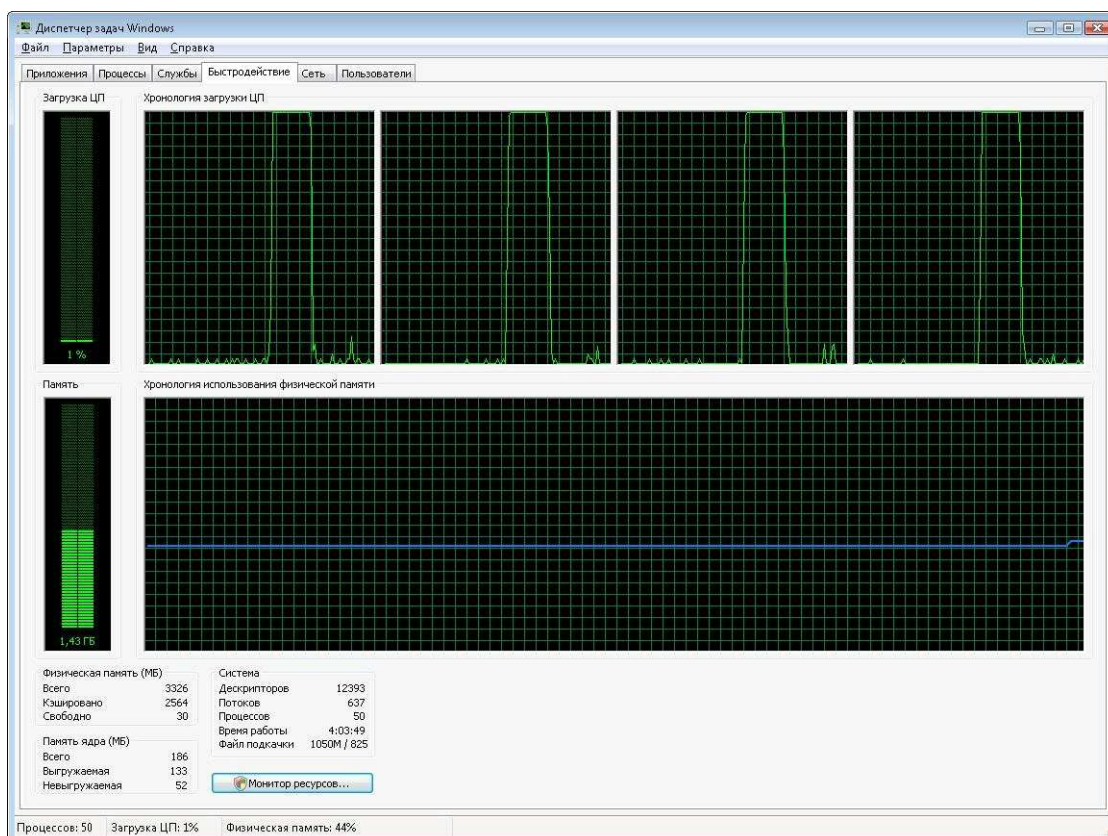
Практика создания алгоритмов показала, что между созданием сложного геометрического объекта и его первым использованием выполняется некоторое количество операторов, в том числе создание и обработка подобных объектов. Таким образом, есть возможность параллельно строить несколько сложных объектов и интерпретировать текст алгоритма до того момента, пока для интерпретатора не понадобится этот объект.

В предыдущей версии системы ЛЕКО для распараллеливания работ использовалась технология распараллеливания на базе OMP. Но его возможностей оказалось недостаточно для эффективной организации параллельного счета поверхностей – выделения больших участков расчета без фиксированного места синхронизации (даже с использованием опции !\$OMP TASKS). Использование OMP для небольших расчетных участков практически не дает ускорения.

Поэтому решено было перевести задачу на использование потоков (Thread), так как они достаточно независимы при исполнении, для потоков отдельно производится квантование времени, они управляемы и есть различные примитивы синхронизации. Для сокращения издержек на создание-завершение потоков, при инициализации задачи создается массив потоков, для которых формируются необходимые данные. Запускаются-останавливаются потоки средствами ОС (SuspendThread, ResumeThread). Так как синхронизация при помощи примитивов ОС в режиме ожидания также потребляет вычислительные ресурсы, то использование синхронизации было сведено к минимуму. Введена критическая секция для запуска-остановки потока. Синхронизация по данным сделана путем выделения для каждого потока отдельного мьютекса, причем, если данные потоком подготовятся раньше, чем они будут востребованы основным потоком, то для синхронизации мьютекс не будет использоваться. Применение мьютекса позволяет синхронизировать основной поток с потоком рассчитывающим данные – остановить основной поток на уровне ОС, поставить его в ожидание конкретных данных от конкретного потока (WaitForSingleObject) и при их готовности продолжить счет максимально быстро. Потоки после расчета переданных данных выставляют признак готовности данных, запускают очередной подготовленный поток из очереди на расчет и сами себя останавливают, таким образом, диспетчеризация и мониторинг пула потоков со стороны основного потока не требуется.

Каждый поток обчисляет свою трехмерную поверхность и свою операцию с этой поверхностью. В основном это итерационный расчет на матрице, с расчетом несложного арифметического функционала. При разработке функционала, обеспечивающего заданные свойства поверхности, ставилась задача не использовать условные операторы, для полного использования конвейера процессора и векторных команд (распараллеливание на уровне выполнения операторов). Требуемая память для расчета параметров одной поверхности - около 200-300 кб. Таким образом, для ускорения расчетов, чтобы не происходило перезагрузки памяти кеша, необходимо держать ограниченное количество потоков в работающем состоянии. Это количество зависит от размера памяти кеша и количества ядер процессора. То есть для начального выбора количества активных потоков можно взять минимальное из чисел $\text{размер_кеша_мб}/0.2$ и количество_ядер . В зависимости от особенностей поверхностей это число можно будет изменять. Например, для CoreQuad Q9650 число одновременно выполняемых потоков может быть выбрано из расчета: от 4 потоков исходя из количества ядер до 60 потоков исходя из ограничений по

размеру памяти кеша. Для i7 с Hyper Treading - от 8 до 40. Эти соотношения позволяют рассчитывать на линейное повышение производительности (сокращение времени реакции) на компьютерах с 24-32 ядрами на одном чипе без возникновения конкуренции по использованию кеша. Массив исполняемых и ожидающих потоков можно устанавливать в 2-5 раз больше, чем количество одновременно исполняемых потоков, и при заполнении всего массива подготовленными к исполнению потоками останавливать выполнение основного потока, обрабатывающего интерпретатор алгоритма-программы.



Пример загрузки четырехядерного процессора - 100% при расчете, сокращение времени расчета в 4 раза.

Дополнительно для осуществления синхронизации по доступу к нескольким видам выделяемой типизированной памяти (списки полей, записей, строк, память для массивов двумерных и трехмерных точек) введены отдельные критические секции для каждого типа памяти, что уменьшает состоятельность потоков. Все объекты используют динамическое распределение памяти, так как размеры объектов могут отличаться в тысячи раз. Для хранимых объектов память выделяется и перераспределяется из общей кучи средствами API (HeapAlloc, ReAlloc), а для промежуточных данных - средствами Фортрана (ALLOCATE и автоматические массивы) и освобождается при завершении расчета очередной поверхности.

При завершении работы интерпретатора – барьерная синхронизация завершения всех потоков. Так как все потоки должны быть завершены в конце работы интерпретатора, а результаты работы некоторых потоков могут понадобиться раньше, поэтому потокам присваивается более высокий приоритет выполнения по сравнению с основным потоком. Если основному потоку необходимы данные потока из очереди потоков, ожидающей выполнения, то такой поток ставится на выполнение, а часть выполняющихся потоков приостанавливается. Остановка потока средствами ОС является самой

экономичной и прозрачной для вычислений операцией. Такая организация вычислительного процесса накладывает минимальные ограничения на дисциплину программирования подпрограмм-потоков, оставляя все возможности применения опций компилятора по оптимизации расчетов.

Каждая трехмерная поверхность - это точки границы, точки сетки, нормали по границе и в каждой точке, треугольная сетка, параметры построения (внутренние линии, точки, нормали на границе). Для разворачиваемых поверхностей это дополняется плоскими (UV) координатами, плоской сеткой более высокого разрешения, условия развертки (посадка – оттяжка участков). Развернутая поверхность может быть визуализирована на место исходной поверхности – т.е. плоская сетка высокого разрешения устанавливается на место (по границе) исходной сетки и моделируются основные свойства ткани (локальная визуализация), т.е создается в памяти копия сетки высокого разрешения с другим расположением узловых точек.

В настоящее время, все описанное выше в плане трехмерной графики с точки зрения программирования реализовано. Идет окончательный выбор и оптимизация алгоритмов формирования поверхностей (их должно быть несколько для различных условий применения). Плоскостная часть программы работает в обычном оптимизированном режиме. Последняя версия программы всегда выложена для свободного скачивания на сайте. После «фиксации» выбранных алгоритмов формирования объектов, описание новых команд языка версии передаются зарегистрированным пользователям и не меняются уже все оставшееся время эксплуатации программы (алгоритмы формирования плоских сплайнов используются уже 18 лет, со времен версии для ДОС с 640 кб памяти, соответственно лекала, разработанные пользователем 15 лет назад будут строится точно так же в самой последней версии программы).

Для разработки и отладки системы использовались компьютеры на базе четырех ядерных процессоров CoreQuad Q9650, 4 ГБ, NVIDIA GTX 280, дисковый массив Velociraprot Raid0, целевой платформой являются компьютеры на базе топовых процессоров i7, Xeon и на базе Larrabee.

Дальнейшее развитие

На этапе разработки сложных многослойных изделий и полной визуализации швейного изделия с более детальным учетом свойств ткани возможно использование нескольких компьютеров, объединенных в сеть. При этом описанное разбиение вычислительного процесса на потоки не меняется. Просто часть потоков будет вести не самостоятельный счет, а при помощи именованных каналов передавать данные на клиентские машины, с запущенной на них локальной программой обсчета поверхностей. Такой поток будет ожидать рассчитанных данных, выделять память под результаты расчета и копировать туда данные. При этом основная машина-сервер производит интерпретацию алгоритма в основном потоке, ведет основной расчет и формирует локальные задания для клиентских машин, ожидающих данных для расчета.

Можно предусмотреть переключение режимов работы основной версии программы: на все компьютеры устанавливаются одинаковые версии системы (соответствующие типу процессора, сейчас пользователю даются версии программы для обычных и топовых процессоров), которые работают или в обычном интерактивном режиме, или если пользователь не работает с программой, он может переключить ее в расчетный режим. Сервер и клиент синхронизируются, создаются именованные каналы в пуле потоков и расчетные

возможности сервера увеличиваются. Среднее время расчета поверхности в зависимости от операции составляет 0.1-1 сек, то есть даже с учетом латентности и скорости передачи сети, разделение задачи на несколько машин выглядит логичным.

Такое решение будет экономически эффективно на небольших предприятиях, имеющих 2-4 быстрых машины и гигабитную сеть. Оно позволит в случае необходимости временно объединять вычислительную мощность компьютеров для решения ресурсоемких задач. Доработка подобного интерфейса межмашинного и межпроцессорного взаимодействия намечена на ближайшее время.

Это решение можно рассматривать как некий «антикластер», в его сегодняшнем понимании. Кластерная технология высокопроизводительных вычислений выработала некоторую традицию к организации вычислений: удаленный доступ для пользователей, пакетный режим, выделенный файл-сервер, администратор и квалифицированный сопровождающий персонал, отдельное помещение, дополнительное кондиционирование, однотипные компьютеры с одинаковыми процессорами, управляющий компьютер, специализированная сеть, дополнительное ПО. Все это идет от больших компьютеров и больших задач, которые считаются сутками и остановка системы или единичный сбой оказываются дорогими во всех отношениях. Это можно рассматривать как перенос высокопроизводительных вычислений «сверху-вниз», со всеми традициями и сложностями больших машин.

Развитие нашей системы конструирования планируется путем увеличения вычислительной мощности «снизу-вверх» - объединение мощностей временно свободных персональных компьютеров рабочей группы для сокращения «времени отклика» системы. Т.е. гибкое подключение-отключение компьютеров по команде пользователей, назначение любого компьютера в качестве главного. Убрать атрибуты кластера: питание, помещение, кондиционер – все как у всех нормальных пользователей ПК, работа без сопровождения и специально обученного системного администратора (предыдущие версии нашей программы работали без вмешательства администраторов по 4-6 лет), убрать «квалифицированные кадры» свести стоимость обслуживания к минимальным значениям, отсутствие специализированного ПО, только ОС с поддержкой сети.

В плане оборудования – типовые высокопроизводительные системы – одно-двух процессорные многоядерные рабочие станции с быстрой памятью, с гигабитной сетью, емкими локальными дисками. Машины можно дополнительно комплектовать 2-4 портовыми гигабитными сетевыми картами и логически объединять их для повышения пропускной способности сети. Стационарные компьютеры могут быть дополнительно соединены сетью напрямую, что уменьшит латентность. Такой переход «снизу-вверх» к высокопроизводительным вычислениям экономически доступен гораздо большему числу предприятий, чем скачок на уровень «современного кластера», даже в исполнении одного шкафа. Такой подход оправдан экономически, так как могут покупаться компьютеры с наилучшим соотношением производительность/цена/доступные средства, а не устаревшие конфигурации для совместимости со старыми системами или замена деталей для всего парка компьютеров. Апгрейт компьютеров может делаться постепенно и исходя из экономической целесообразности. Обычные рабочие станции более универсальны: если не требуется расчет, то можно выполнять любую другую работу. Вместе с тем, производительность (на наших задачах) будет на уровне суперкомпьютера, так как накладные расходы в рамках одного компьютера и в рамках сети минимальны, а оптимизирующий компилятор позволяет выжать максимальную производительность из каждого конкретного

процессора (кстати, критерием для оценки работы системы будет не характерная для суперкомпьютеров характеристика «время отклика задачи»).

На первый взгляд может показаться, что это «не серьезный подход». Но если взять характеристики многомашинных кластеров установленных 4-5 лет назад, то окажется, что возможности этих дорогих кластеров перекрывают 2-3 современные рабочие станции практически по всем параметрам. И для работы этим рабочим станциям нужна «только розетка». Пользователи давно эксплуатируют многоядерные компьютеры, даже иногда не подозревая об этом. На таком же уровне должны работать задачи с распределенным расчетом – прозрачно для пользователя распараллеливаться на уровне команд процессора, потоков, вычислительных ядер, процессоров, и деле машин, при минимальных действиях со стороны пользователя. Для пользователя использование ресурсов соседних компьютеров должно быть на уровне использования своего компьютера - нажал на кнопку и программа стала выполняться быстрее, ушел сосед на обед - программа работает еще быстрее. И это без администратора, инженера, системного программиста и консультанта. Если такое недостижимо в настоящий момент для всех задач, то для нашей задачи такая реализация вполне возможна.

Еще раз о задаче

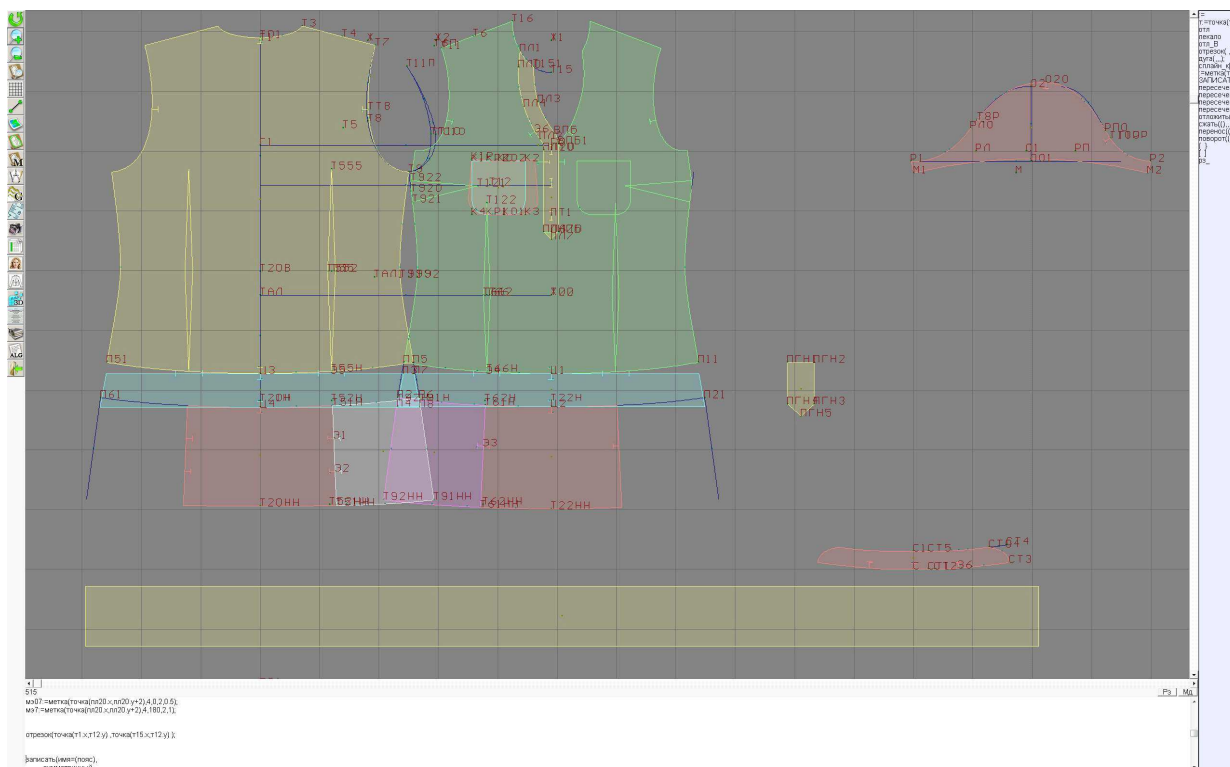
До сегодняшнего времени практически не существовало систем для трехмерного проектирования одежды (известные образцы относятся к прообразам систем, а не рабочему инструменту конструктора). Система может быть работоспособной при времени реакции в процессе разработки лекал до 10 сек, так как разработка ведется не только алгоритмически, но и визуально. Расчет окончательного варианта может занимать больше времени, но все равно должен исчисляться минутами, а не часами.

Особенностью разработки лекал одежды является то, что лекала одежды должны оцениваться комплексно, т.е. нельзя по внешнему виду одной детали оценить ее качество. На детали не существует какой-либо документации или нормативов, швы и конструктивные точки могут перемещаться конструктором в пространстве, объемы перераспределяться и задача конструктора состоит в поиске некоего компромисса, как из плоских деталей создать объемное изделие соответствующее фигуре, обеспечивающее заданный уровень функциональности, комфорта и выполняющую эстетическую функцию. Как правило, конструктор создает несколько вариантов изделия, или вносит изменения в текущий разрабатываемый образец. В целом, результат работы конструктора всегда оценивается субъективно, например, художественным советом, и опосредовано – оцениваются не плоские лекала, а изделие, отшитое из конкретной ткани по заданной технологии одетое на манекен или на человеке. При этом дефекты посадки и технологические дефекты определяются вполне объективно.

Отсутствие программ трехмерного конструирования одежды было связано с недостаточной производительностью компьютеров (и сейчас производительность обычных персональных компьютеров недостаточна для полного решения этой задачи). При падении цен на компьютеры достаточной производительности (около 1-2 тф) до цен уровня 3-4 тыс. долларов сектор программ трехмерного конструирования одежды и сектор использования таких программ будет активно развиваться.

С другой стороны существует спрос со стороны конечных потребителей, которые готовы воспользоваться результатами трехмерного конструирования лекал. При этом они могут использовать обычные компьютеры, т.к. ожидание 5-7 мин и более не является для них критичным. Для обычных пользователей так же

актуальна задача параллельного счета: у многих не один компьютер, есть проводная гигабитная сеть и сократить время счета с 8 минут до 4 никто из них не отказался бы.



Немного о востребованности систем проектирования и конечного результата – выкроек. Востребованность выкроек, построенных на основе трехмерного конструирования можно оценить на основе востребованности результатов предшествующей версии с более простым плоскостным конструированием. При успешной разработке описанной трехмерной системы возможно расширение области применения, а именно: головные уборы, обувь, мягкая игрушка, воздушные шары и надувная рекламная продукция, сумки, туристическое снаряжение, спецодежда по индивидуальным размерам (например, костюм для дайвинга). Плоскостная версия системы ЛЕКО уже давно используется фирмой «Кентавр-наука» для проектирования используемых в полетах комбинезонов для российских космонавтов.

Обсуждения конкурирующих решений не будет, так как о трехмерном конструировании материалы публиковали только фирма Стаприм (уже не существует) и Ассоль. Система Ассоль работает на базе Автокада и судя по описаниям содержит большое количество ручной подготовки данных. Зарубежные фирмы публикуют еще меньше материалов, работы в основном старые, иллюстрации не дают возможности оценить работоспособность системы. Оценивать систему трехмерного конструирования лекал можно только тогда, когда на этой системе реально разрабатываются лекала.

Из зарубежных наиболее зрелищные результаты у Оптитекс (Израиль), Browzwear (Израиль), но это визуализация изделий по готовым лекалам, а не разработка лекал. Скорость появления иллюстративных материалов – 10 роликов за 3 года, говорит о наличии сложностей. Из программ выдающих лекала – Дижитал Фешн (Япония) – сильно ограниченная возможность формирования поверхностей. В Европе финансировалась программа по внедрению цифровых технологий в швейной отрасли E-Тейлор, которая закончилась неопределенным

отчетом о необходимых направлениях разработки. Ссылки на сайты приведены ниже.

Хороший пример дает инициативная группа конструкторов на сайте www.season.ru, которые пытаются решить задачу трехмерного конструирования лекал используя различные 3D пакеты (Автокад, Рино, Даз, Позер, Дижитал Фешн и т.д.): сотни страниц обсуждений, реальные работы, фотографии и обучающие ролики. Там же разбираются все сложности и недостатки при использовании этих пакетов. Ни один из них не дает возможность построить лекала без значительной ручной работы над каждым комплектом лекал.

Разработчики «швейных» САПР

Зарубежные: Лектра, Гербер, Инвестроника, Тукатех, Ассист, Графис, менее распространенные системы есть в Болгарии, Румынии, Китае, Сингапуре

Отечественные: Вилар софт, Комтенс, Ассоль, Стаприм (сайт уже не работает)

Украинские: САПРЛЕГПРОМ, Грация

Ссылки на сайты разработчиков швейных САПР

www.latelye.ru, www.lekala.info, www.comtense.ru,
www.tflex.ru/products/tfdres01.htm, www.julivi.com, www.cadrus.ru, www.assol.mipt.ru,
www.saprgrazia.com

www.lectra.com, www.gerbertechnology.com, www.optitex.com,
www.browzwear.com, www.coat.de www.cadcam.solutionsaustralia.com.au
www.wildginger.com www.polytropon.com

Самые популярные русскоязычные «швейные» сайты-форумы (до 30000 человек в день)

www.osinka.ru, www.season.ru

Кратко

коэффициент эффективности распараллеливания

Изложенный подход сейчас обеспечивает эффективную загрузку многоядерной и многопроцессорной системы с общей памятью, обеспечивая практически линейное увеличение производительности (сокращение времени отклика) до 24-32 ядер без увеличения накладных расходов (ожидание на синхронизацию и конкуренции за кеш-память). Под словом «линейное» понимается линейное с коэффициентом близком к 1.

Ставится задача организации многомашинного счета доступного и прозрачного для пользователя.

обоснование востребованности использования высокопроизводительных суперкомпьютеров (HPC) для решения выбранной задачи

Обычный компьютер поставленную задачу не решает. Интерактивная работа конструктора требует быстрого расчета, что в ближайшее время реализуемо только с использованием технологий суперкомпьютеров.

абсолютная производительность

использование компилятора Intel(R) Visual Fortran Compiler Professional Edition 11.1 в сочетании с VTune позволяет в полной мере использовать все возможности современных процессоров, системы команд и расширения до SSE4.2.

сложность приложения

Сложность приложения подтверждается тем, что не смотря на то, что такие системы востребованы, они еще не реализованы. Основная сложность – алгоритмы, принципы работы, согласование их с доступной архитектурой и возможностями компьютеров.

оригинальность реализации

15 лет назад система ЛЕКО была единственной параметрической системой конструирования одежды. Сейчас параметризация построения становится нормой. Реализованная у нас масштабируемость вычислений, использование многоядерности, ориентация на многомашинный счет в описании других «швейных» САПР даже не упоминается. Постановка задачи использования нескольких машин, динамически объединяемых для ускорения расчетов в готовой для реализации форме, является достаточно новым взглядом на высокопроизводительные вычисления.