1. この研究の目的は，小傾角粒界エネルギーにおけるRead-Shockleyの理論予　　測と大槻の実験結果の相違点を解明することである．
2. 両者の結果の相違点を，具体的に図を用いて説明していきます．
3. まず，Read-Shockleyの理論予測を支持する結果になった，Hassonらによるシミュレーションは，図(a)のように，角度，０度及び90度付近の傾き方が異なって表示された．
4. その一方で，大槻による実験結果では，図(b)のように，角度０度，及び90度の傾き方が左右同じになり，二つの結果に矛盾が生じた．
5. この矛盾を解明するために，西谷研究室では様々な手法をこれまで取り組んできた．
6. まず，八幡の研究では，２種類の原子間ポテンシャルを使ったシミュレーションをおこない，その結果をもとにSuttonVitekの粒界モデルを用いたVASPで粒界エネルギーを計算した．
7. その結果，図２のように，０度及び９０度の傾き方が異なるグラフ，すなわちRead-Shockleyの理論予測を支持する結果になった．
8. これは，粒界エネルギーの計算において安定した原子構造を配置できなかったことが原因である．
9. この原因をもとに，岩佐の研究では，最安定な原子配置を探索するための原子削除操作を取り入れ，系全体のエネルギーを計算した．
10. その結果，粒界エネルギーが予測通り低い値となり，大槻の実験結果を支持する形になったが，
11. 原子配置を表示した際，図(a)のように，４個原子を削除した時は安定した構造になっているのに対し，６個削除した時は, (b)のように，原子全体が移動してしまい，構造緩和に過ちが生じていた．
12. これは，外部緩和をおこなった際に粒界がより低い角度になった状態で計算をしてしまったことが原因である．
13. この原因で構造緩和に過ちが生じたのは，岩佐が安定構造の原子配置を視覚的に確認しなかったためであるので，本研究で原子配置を簡易的に視覚化できるように開発していく．
14. その開発についてですが，web application開発の手法の一つ“MVCモデル”で作成していく．
15. MVCモデルは，データの処理をおこなう”Model”，処理結果を画面表示する”View”，この二つに対し指令を出す”Controller”で構成され，それぞれ機能が分離されている．
16. 私がおこなうのは，この“view”の機能であり，研究室内で開発されたモデル”Viewer”を使用して処理結果を出力していきます．
17. "Viewer"は，小傾角粒界の原子モデルを視覚的に確かめるためのモデルであり， POSCAR形式のファイルを入力とします．
18. 本研究で開発して，この”viewer”をSVGで出力できるようにしていく．なお，SVGの出力は，２次元画像描画ライブラリ“rcairo”を使って作成していきます．
19. 今後の課題ですが，このプロトタイプをもとに，より簡易的に視覚化できるよう研究を進めていきます。
20. 具体的には、原子配置の見やすさの向上，並びに“SVGスプライト”という機能を用いて複数の配置を表示・比較できるように開発していく．

（POSCARというのは，第一原理計算ソフトVASP における原子の位置が入力されたファイルのことです．）

（rcairoは，入出力する際の画像形式に固定されず同じ描画処理を使用できるため，その状況に応じて形式を変更し描画することができる．）

（そして，完成したソフトを使って第一原理計算ソフトVASPによる粒界エネルギーの計算を進め，二つの結果の矛盾を解明する研究をおこなう予定．）

研究のコア

“Viewer”をSVGで出力することにより，原子配置を見やすくて画像表示しやすいソフトに開発．

※viewer のもとの出力はPNG(ドットマップ画像)．

**・なぜSVGで出力することが視覚化しやすいといえるのか？**

SVGで出力することにより，図の拡大・縮小の際に画像の補完がおこなわれ，画質が乱れることなく表示することが可能になります．これにより，原子配置が正しい位置でシミュレーションできているかを視覚的に確認しやすくなる．

また，レイヤー機能もサポートされているため，画像データの表示・交換を簡単に処理でき，〜．

他のフォーマットと比べて，小さい容量で画像ファイルを作成でき，

また，画像形式だけでなく，XMLで記述されているテキストデータであるため，テキストエディッタでも画像の作成ができる．

→XMLで書かれるため，これまでの実験結果における各画像のデータに情報を書き込めて大量のデータを一括管理または分類することができる．

また，特定のデータを集計することもでき，XMLを利用してwebアプリのデータと連動することも容易にできる．

**・MVCモデルのModel, Controllerにあたる機能は誰がどのようにおこなうのか？**

**・“viewer”の入力ファイルであるPOSCAR.txtは，岩佐の原子削除操作をおこなったものをしようするのか？**

**・SVGで出力する具体的なアクションは何か？（png→svg）**

* Sutton Vitekの粒界モデル
* 回転角をつけた２つの完全結晶の間に，３層構造から１層構造に変化する層（buffer layer）

・cairoを用いる利点

可能な限り同一の出力をサポートし，画面のコンテンツを同じライブラリで描画し，高解像度な画面で出力できる．

描画コンテキストはパスを管理するだけではなく、変換行列も管理 しています。この変換行列というものを使えば、パスを作る部分を 変更することなく、パス全体を移動させたり、回転させたり、拡大 縮小したりできます。

また，比較的大量の情報でも比較的小さな方程式にエンコードできるため，cairoでベクトル画像を描画すると，小さいサイズで作成が可能．

その上，点や線などのそれぞれに関連づけられた方程式に従って作成するため，正確かつ描画の単純化になる．

cairo:（→描画操作でサーフェスを意識する必要がない，すなわち画像ファイルの形式が入出力で異なっても同じ描画処理を使い回すことができる．（　png→svg））

rcairoは，Ruby言語にあわせたAPIでcairoを使用する．

欠点：rcairoは，拡張ライブラリなので通常のライブラリよりも導入が大変．

＜rcairo で画像作成する手順＞

1. サーフェスを作成

2. 作成したサーフェス用のコンテキストを作成

3. コンテキストに対して描画処理する

4. サーフェスに対して終了メッセージを送る

※サーフェスは出力先を抽象化したオブジェクト

　描画操作はコンテキストがおこなう

・MVCモデル

ソフトウェア設計のモデルの一つ．

＜具体的に，“Controller"はユーザからのリクエスト情報を受け取り，それを基に”Model"に対して処理を依頼する．“Model"はデータと連携して処理を行い，その処理結果を”Controller"に返す．”Controller"は返ってきた処理結果データを”View"に渡し，”View"はデータを基にHTML出力処理を行う．＞

・SVGスプライト

あらかじめ描画するオブジェクトをひとまとめにして必要なときに呼び出せる機能．

pngは様々な色やサイズの画像を必要な分だけ作成する必要があるが，

SVGスプライトだと一つのファイルで様々なバリエーションに応じて使用が可能．

グラフィックに変更があった場合でもcssで対応ができる．

欠点：グラフィックを文字列に置き換えている分，通常のhtmlの要素に比べて記述量が多くなる．

作成方法：illustratorでアートボードを設定し，描画したものを保存し別のファイルで書き出す．SVGファイルを呼び出してcssで装飾すれば，描画したものをすぐに呼び出せる

第一原理計算ソフトVASP = 粒界モデル内における原子の安定位置や系の

全体エネルギーを出力するソフト