TECHNISCHE UNIVERSITÄT MÜNCHEN

Max-Planck-Institut für Biochemie
Abteilung für Molekulare Strukturbiologie

Verfahren zur Klassifikation von Partikeln in elektronenmikroskopischen Tomogrammen

Michael Stölken

Vollständiger Abdruck der von der Fakultät für Chemie
der Technischen Universität München
zur Erlangung des akademischen Grades eines
Doktors der Naturwissenschaften (Dr. rer. nat.)
genehmigten Dissertation.

Vorsitzender:	Univ.-Prof. Dr. Johannes Buchner
Prüfer der Dissertation:	1. Hon.-Prof. Dr. Wolfgang Baumeister
	2. Univ.-Prof. Dr. Sevil Weinkauf

Die Dissertation wurde am 25.01.2012 bei der Technischen Universität München eingereicht und durch die Fakultät für Chemie am 02.04.2012 angenommen.

Meiner Mutter

Teile dieser Arbeit wurden im folgenden Artikel veröffentlicht:

Maximum likelihood based classification of electron tomographic data. M. Stölken, F. Beck, T. Haller, R. Hegerl, I. Gutsche, J.M. Carazo, W. Baumeister, S.H.W. Scheres, S. Nickell. Journal of Structural Biology, 2011;173(1):77-85.

Inhaltsverzeichnis

Abbildungsverzeichnis

Zusammenfassung

Verfahren zur Klassifikation und Mittelung von Subtomogrammen können die Auflösung in Strukturuntersuchungen molekularer Komplexe verbessern und ermöglichen damit eine Analyse der strukturellen Merkmale mit höherer Präzision. In dieser Arbeit wird ein neu entwickelter dreidimensionaler (3D) *Maximum Likelihood*-Algorithmus im Vergleich zu bestehenden Analysemethoden vorgestellt. Der ML-Algorithmus fasst die Alignierung und Klassifikation in einem einzigen Prozessschritt zusammen und verwendet ein neu-definiertes Abstandsmass, welches auch für andere Verfahren einsetzbar ist. Dieses Abstandsmass bildet die Grundlage für die Berechnung einer Wahrscheinlichkeit, die prüft ob ein individuelles Subtomogramm zu einer gegeben Referenzstruktur (bzw. Klasse) gehört. Dabei wird angenommen, dass die Referenzstruktur durch einen *Compound Wedge* verzerrt ist, der sich aus unterschiedlich orientierten *Missing Wedge* Volumen zusammensetzt. Das Abstandsmass berücksichtigt den *Compound Wedge* der Referenz und den *Missing Wedge* des einzelnen Subtomogramms und gewichtet die Frequenzbereiche von Subtomogramm und Referenz beim Vergleich entsprechend. Um die Berechenbarkeit dieses Verfahrens auch für große Volumen mit einer großen Anzahl an Subtomogrammen zu gewährleisten, wurde MLTOMO entwickelt. Das C++ Programm parallelisiert den *Maximum Likelihood* (ML)-Algorithmus und ermöglicht eine skalierbare Berechnung des Problems auf Super-Computern mit einer großen Anzahl an Prozessoren. Simulationen zeigen, dass MLTOMO präzisere Ergebnisse liefert als der *Constrained Correlation* Ansatz. Das Verfahren zeigt insbesondere Vorteile für die Analyse von Strukturen, die in einer bevorzugten Orientierung in der Probe und damit in den Subtomogrammen vorliegen. Bei der Untersuchung von kryo-elektronenmikroskopischen Daten (in Eis eingebettete Thermosomen), konnten mit Hilfe von MLTOMO zwei unterschiedliche Konformationen des Thermosoms identifiziert werden. Die Ergebnisse zeigten darüber hinaus große Übereinstimmung mit vergleichbaren Untersuchungen aus Einzelpartikelanalysen, denen Proben mit derselben Präparation zu Grunde lagen.

1 Einleitung

Kryo-Elektronentomographie ist ein bildgebendes Verfahren, dass es ermöglicht 3D-Rekonstruktionen von biologischen Proben zu erstellen und birgt damit ein großes Potential für die molekulare Zellbiologie [McIntosh, 2001] [Frank et al., 2002] [Lucic et al., 2005]. Große, supramolekulare Strukturen können so *in situ* in zellulären Umgebungen untersucht werden. Allerdings ist das Signal-Rausch-Verhältnis in Kryo-Elektronentomogrammen, durch die Strahlempfindlichkeit der Proben und die notwendigen Niedrigdosisbedingungen, niedrig. Bedingt durch den experimentellen Aufbau und die Aufnahmegeometrie fehlen zusätzlich Bildinformationen in gewissen Bereichen der zu untersuchenden Struktur. Dies führt zu einer nicht isotropen Auflösung in den resultierenden Tomogrammen. Beide Effekte erschweren die Analyse der tomographischen Daten.

Analyseverfahren aus dem Bereich der Einzelpartikel-Analyse [Frank, 2002] können bis zu einem gewissen Grad für die Klassifikation tomographischer Daten herangezogen werden. Auch wenn diese Methode sich nicht direkt auf pleomorphe Strukturen wie Zellorganellen oder ganze Zellen übertragen lassen, so enthalten die Tomogramme oft repetitive Komponenten. Diese Strukturen können *in silico* aus den Tomogrammen extrahiert und analysiert werden. Durch die Mittlung von N einzelner abgebildeter gleicher Strukturen X_i mit $i \in \{1, 2, ...N\}$ kann das Signal und damit die Auflösung einer gemittelten Struktur A verbessert werden. Vorausgesetzt, die Strukturen werden vor der Mitteilung in einen Alignierungsschritt durch die Transformation R_{ϕ_i} gleich ausgerichtet:

$$A = \frac{1}{N} \sum_{i=1}^{N} R_{\phi_i} X_i. \tag{1.1}$$

Die Einzelpartikel-Analyse nutzt aus, dass gleiche Strukturen in der Probe zufällig orientiert sind. In den elektonenmikroskopischen Aufnahmen entstehen Abbildungen, die zweidimensionale (2D) Projektionen der Strukturen aus unterschiedlichen Richtungen darstellen. Es benötigt viele unterschiedliche Orientierungen der gleichen Struktur, um ein dreidimensionales Bild zu rekonstruieren.

Bei der Elektronentomographie haben die Projektionen einer Struktur eine wohl definierte Orientierung zueinander. Somit lässt sich durch Überlagerung der Projektionen eine 3D Rekonstruktion jeder einzelnen Struktur erzeugen. Aufgrund der geometrischen Anordnung des Probenhalters zum Elektronenstrahl ist nur ein Bereich von $\pm 70°$ für die Datenaufzeichnung zugänglich. Somit fehlen in bestimmten Bereichen der Tomogramme Daten.

Betrachtet man den Fourierraum des abgebildeten Objekts entsteht bei der Einachsenkippung eine keilförmige Datenlücke. Dieser Bereich, in dem keine Daten vorliegen, wird häufig als *Missing Wedge* bezeichnet und beträgt 20-30% des gesammten Volumens. Eine Doppelachsen-Kippung kann den Bereich an fehlenden Daten auf 10-15% vom Gesamtvolumen reduzieren (*Missing Pyramid*), indem eine zweite Kippserie nach einer Rotation der Probe in der Ebene um 90° aufgenommen wird. Generell führt die fehlende Information in den rekonstruierten Tomogrammen im Fourierraum zu einer anisotophen Auflösung der Abbildung im Realraum und erzeugt unter anderem eine Elongation der abgebildeten 3D-Strukturen.

Schwerpunkt dieser Arbeit ist die Entwicklung von Klassifikationsverfahren, die eine Interpretation der tomographischen Daten vereinfachen oder erst ermöglichen. Dafür wurden mathematische Modelle erstellt, die im Kern auf Mittelungsverfahren (1.1) basieren und unter anderem die beschriebenen Effekte berücksichtigen. Die Modelle sind die Grundlage für ein erstelltes skalierbares Computerprogramm, dass die Berechenbarkeit der Analyse auch für Tomogramme mit großen Datenmengen gewährleistet.

2 Elektronentomographie

Die grundlegende Technik der Elektronentomographie ist die Aufnahme zweidimensionaler Projektion, eines biologischen Objekts aus unterschiedlichen Richtungen mit einem Transmissionselektronenmikroskop (TEM). Im Vergleich zum Lichtmikroskop wird die Abbildung der Probe durch die Interaktion mit Elektronen anstelle von Photonen erzeugt. Um die Absorption oder Streuung der Elektronen durch Luftmoleküle zu vermeiden, müssen die Untersuchungen im Hochvakuum durchgeführt werden. Biologische Proben bestehen in der Regel aus 70% bis 80% Wasser. Daher ist eine spezielle Probenpräparation notwendig, um die plötzliche Verdampfung des Wasser im Vakuum zu vermeiden. Während der Datenaufzeichnung wird die Probe gekippt, um Aufnahmen von Projektionen aus unterschiedlichen Richtungen an der gleichen Stelle zu erhalten. Durch tomographische Rekonstruktion erhält man eine dreidimensionale Abbildung der untersuchten Strukturen. Aufgrund der Strahlenenpfindlichkeit biologischer Proben können nur sehr geringe Elektronendosen verwendet werden. Demzufolge ist das Signal-Rausch-Verhältnis der Einzelprojektionen und der resultierenden Tomogramme sehr niedrig. In den meisten Fällen bedarf es Verfahren der Bildverarbeitung und Mustererkennung um makromolekulare Strukturen in der tomographischen Rekonstruktion zu identifizieren.

2.1 Probenpräparation

Um biologische Proben für die Untersuchung im Hochvakuum des TEM zu präparieren, kann der Probe entweder das Wasser entzogen (durch chemische Fixierung), ersetzt (durch Einbettung in Kunststoff) oder eingefroren werden (durch Kryopräparation).

Bei der chemischen Fixierung wird die Probe getrocknet und mittels Schwermetallsalzen kontrastiert [Palade, 1952]. Die Bereiche der Probe, die kein biologisches Material enthalten, werden bei diesem Verfahren mit einer dünnen Salzschicht

überzogen. Nach Probentrocknung wird das Salz und das biologische Material ausgewaschen und es verbleibt ein Negativabdruck der Strukturen. Die Auflösung ist beschränkt auf die Größe der Salzkristalle ($\sim$ 2 nm) und die Methode verändert die Struktur der Probe. Beides erschwert die Interpretation auf molekularer Ebene.

Die Einbettung der Probe in Kunststoff ist ein weiteres Verfahren zur Probenpräparation, bei dem die Präparate in flüssigen Kunststoff eingebettet werden, der schließlich aushärtet. Es wird häufig verwendet, um Schnitte durch grössere Objekte beispielsweise Gewebe herzustellen, die sonst für elektronentomographische Untersuchungen unzugänglich wären [Richardson et al., 1960]. Die „Plastikschnitte“ bieten den Vorteil, dass sie sehr langlebig sind und höheren Strahlendosen ausgesetzt werden können. Aber auch bei diesem Verfahren kann es zu einer erheblichen Veränderung der Struktur der Probe kommen, so dass die Abbildungen der Komplexe stark von ihrer nativen Form abweichen.

Mit der Entdeckung, dass Wasser bei extrem schneller Abkühlung einen direkten Phasenübergang vom flüssigen in einen amorphen Zustand macht, ohne Eiskristalle zu bilden, wurde die Möglichkeit geschaffen, den nativen Zustand der Probe zu erhalten und mit dem TEM zu untersuchen [Adrian et al., 1984]. Bei der so genannten Kryopräparation dient ein Kupfernetz (Grid) als Probenträger, auf dem sich ein löchriger Kohlefilm befindet. Auf den Kohlefilm wird die Molekül- oder Zellsuspension aufgebracht und mit Filterpapier abgesaugt, bis die Löcher nur noch von einem dünnen Film bedeckt sind. Das Grid wird dann in flüssigen Ethan “eingeschossen” und bei einer Temperatur von -190°C vitrifiziert. Dieses Verfahren erhält die nativen Strukturen. Allerdings ist der Kontrast in den resultierenden Aufnahmen gering und die Probe ist sehr strahlenempfindlich. Erschwerend kommt hinzu, dass die Probe permanent auf der Temperatur von flüssigem Stickstoff gehalten werden muss. Das bedeutet, dass die Kryo-Bedingungen während der gesamten Analyse bestehen müssen, sowohl beim Transfer der Probe in das TEM, als auch bei der weiteren Datenaufzeichnung oder der Lagerung der Probe.

2.2 Datenaufzeichnung

Zur Aufzeichnung von 2D-Projektionsbildern wird die biologische Probe nach der Präparation in einen Kipphalter eingesetzt und in das TEM befördert. Der Kipphalter wird während der Aufnahme durch das Goniometer um eine Achse gedreht, dadurch entsteht eine Serie von Projektionen aus unterschiedlichen Richtungen, die

als Kippserie bezeichnet wird. Aus einer Kippserie kann dann die dreidimensionale Massendichteverteilung des Objekts rekonstruiert werden (Abb.: 2.1).

Durch die Geometrie des Probenhalters und die Strahlgeometrie sind Kippwickel von 90° im TEM nicht möglich, da ansonsten der Halter den Elektronenstrahl abschatten oder die Stege des Objektträgernetzchens in das Bild einwandern würden. Somit käme es bei zu hohen Kippwinkeln zu einer kompletten Abschattung der Probe. Außerdem hat die Probe eine große laterale Ausdehnung (Mikrometer) und eine kleine Dicke (wenige 100 Nanometer). Der Weg durch die Probe wird mit zunehmendem Kippwinkel für den Elektronenstrahl immer länger, was bei großen Kippwinkeln das Durchstrahlen der Probe verhindern kann. Aus diesem Grund können bei einer Kippserie nur Projektionen innerhalb eines Kippwinkelbereichs von maximal $-70°$ bis $+70°$ aufgenommen werden.

Eine Rekonstruktion ist nur dann möglich, wenn alle Projektionen an derselben Stelle der Probe aufgenommen werden. Allerdings ist die gesamte Elektronendosis zur Abbildung einer Position in der Probe begrenzt. Die Wechselwirkungen (vor allem inelastische Wechselwirkungen) der einfallenden Elektronen mit der biologischen Materie führen zur Veränderung und letztlich zur Zerstörung der Probenstruktur. Es werden Radikale im Probenmaterial erzeugt, die sofort weiter reagieren. Die chemischen Bindungen werden so zerstört und es kommt zum Massen- und Strukturverlust des biologischen Materials. Bereits das Einzelbild einer biologischen Probe mit der maximal möglichen Dosis ($\sim 50\, e^-/\text{Å}^2$) liefert unter Kryo-Bedingungen nur ein sehr signalschwaches Bild. Vor diesem Hintergrund erscheint es nicht besonders sinnvoll, die limitierte Elektronendosis für eine Probe auf eine Vielzahl von Bildern zu verteilen. Jedoch wurde von Hegerl und Hoppe [Hegerl and Hoppe, 1976] ein Theorem der Dosisaufteilung aufgestellt, das von McEwen [McEwen et al., 1995] experimentell bestätigt wurde. Dieses Theorem besagt, dass ein Tomogramm die gleiche Signalstärke besitzt wie eine einzelne Projektion, die mit derselben Dosis aufgenommen worden ist. Andere Untersuchungen bestätigen zudem, dass der Informationsgehalt in einem Tomogramm sogar höher ist als in einer Einzelprojektion bei gleicher Dosis und es nur davon abhängt, ob die erreichbare Auflösung ausreicht um einzelne Makromoleküle im Tomogramm voneinander zu separieren [Saxberg and Saxton, 1981]. Erschwerend kommt hinzu, dass die Dichte von Proteinen und die Dichte von Wasser nahe beieinander liegen, wodurch der Kontrast in kryoelektronischen Aufnahmen zwischen den Strukturen des biologischen Materials und dem umgebenden vitrifizierten Wasser gering ist. Die Limitierung der Elektronendosis und der geringe Dichteunterschied der Strukturen zum umgebenden Medium sind die wesentlichen Faktoren für einen

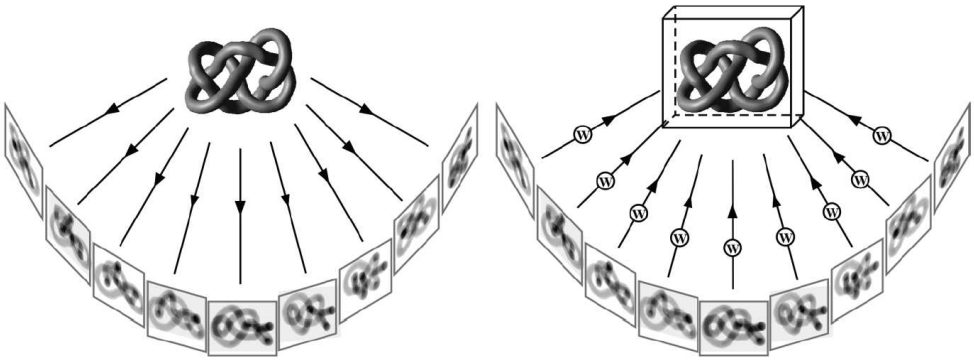


Abbildung 2.1: Prinzip der Elektronentomographie. Die Rekonstruktion eines Objekts aus einer Serie von Einzelprojektionen, aufgenommen aus verschiedenen Richtungen, wird als Tomographie bezeichnet. a) Aufnahme von Projektionen. In einer Kippserie wird die Probe für jede Aufnahme um eine feste Kippachse bis zum maximalen Kippwinkel gedreht. b) Rekonstruktion des dreidimensionalen Volumens. Durch Rückprojektion der Kippserie kann ein dreidimensionales Abbild der Massendichteverteilung der Probe erzeugt werden

geringen Kontrast und damit für ein niedriges Signal-Rausch-Verhältnis in den Aufnahmen.

2.3 Dreidimensionale Rekonstruktion

Die Rekonstruktion basiert auf einem Prinzip, das zuerst von Radon [Radon, 1917] entdeckt wurde: Eine Funktion, die z.B. eine Dichteverteilung repräsentiert, kann auf der Basis ihrer Projektionen rekonstruiert werden. Dieses Prinzip wird in allen Verfahren der Tomographie genutzt und wurde zuerst von DeRosier [DeRosier and Klug, 1969] und Hart [Hart, 1968] in der Elektronenmikroskopie realisiert. Es kann angewendet werden, weil die elektronenmikroskopische Abbildung in guter Näherung eine Projektion des elektrostatischen Potentials der Probe darstellt. Wird die Probe nun als Massendichteverteilung im Raum angenommen, entsprechen die Fouriertransformierten der Projektionen zentraler Schnitte, die in unterschiedlichen Richtungen durch die Fouriertransformierte dieser Massendichteverteilung laufen. Dieser Zusammenhang wird auch als Projektions-Schnitt-Theorem bezeichnet (Abb.: 2.2) [Crowther et al., 1970] [Dudgeon and Mersereau, 1984].

Aufgrund der Objektdicke haben auch die Schnitte durch den Fourierraum eine gewisse Ausdehnung. In Abbildung 2.2 wird deutlich, dass mit steigender Anzahl der Projektionen eine bessere Abdeckung der Strukturinformation im Fourierraum möglich ist. Dies gilt inbesondere für die Strukturinformation bei höheren Frequenzen. Die Grenzfrequenz (Kugelradius im Fourierraum), bei der alle Strukturfaktoren noch zugänglich sind, wird Crowther Kriterium genannt [Crowther et al., 1970]. Es ist ein Maß für die erreichbare Auflösung d_C und kann für eine Rekonstruktion eines sphärischen Objektes in Abhängigkeit von der Anzahl der Projektionen N_p und der Objektdicke D leicht abgeschätzt werden:

$$d_C \simeq \frac{\pi D}{N_P}. \tag{2.1}$$

Jedoch bedeuten mehr Projektionen in der Praxis nicht gleichzeitig eine bessere Auflösung, denn die Elektronendosis über alle Projektionen ist limitiert. Die richtige Balance zwischen der Anzahl an Projektionen und der Elektronendosis für eine Kippserie ist bei der Kyo-Elektronentomographie entscheidend.

Die Schnitte im Fourierraum überlagern sich in der Form, dass die Überdeckung vom tiefen in den hohen Frequenzbereich abnimmt. Ein mögliches Augleichsverfahren zur tomographischen Rekonstruktion ist die gewichtete Rückprojektion. Bei dieser Methode werden die einzelnen Projektionen gemäß ihrer Projektionsrichtung in ein gemeinsames dreidimensinales Volumen zurückprojiziert. Die Rückprojektion erfolgt dabei im Allgemeinen numerisch mit trilinearer Interpolation im Realraum. Dabei wird die Überlagerung der Frequenzen durch eine so genannte Wichtungsfunktion im Fourierraum korrigiert [Hoppe and Hegerl, 1980] [Harauz and van Heel, 1986]. In Abbildung 2.3 sind Schnitte durch eine Rekonstruktion einer Zelle gezeigt, die aus 120 Projektionen errechnet wurde.

Durch den begrenzten Kippwinkel bei der Kryo-Elektronentomographie entsteht ein Bereich, in dem keine Strukturinformation vorhanden ist. Dieser keilförmige Bereich im Fourierraum wird *Missing Wedge* genannt. Bei der Rekonstruktion entstehen durch die fehlenden Projektionen orientierungsabhängige Rekonstruktionsartefakte. Der *Missing Wedge* führt damit zu einer anisotropen Auflösung innerhalb des Tomogramms. Diese orientierungsabhängige Verzerrung ist ein gravierendes Problem, denn die Orientierung der makromolekularen Strukturen in der Probe sind nicht bekannt und oft zufällig verteilt. Gleiche Strukturen können bei einer unterschiedlichen Orientierung zur Kippachse unterschiedliche Massendichteverteilungen aufweisen (Abb.: 2.4). Die eindeutige Identifizierung der Strukturen erfordert aber gerade eine möglichst orientierungsunabhängige Rekonstruktion.

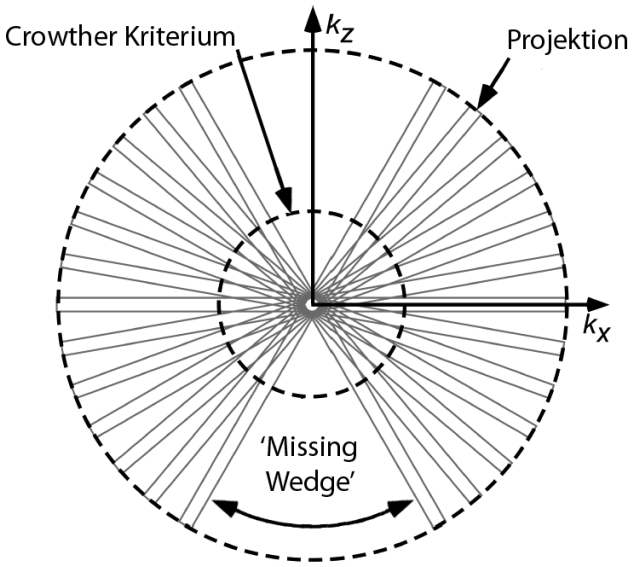


Abbildung 2.2: Zweidimensionale Veranschaulichung des Crowther Kriteriums. Die Fouriertransformierte jeder Projektion ist als Schnitt durch den Fourierraum dargestellt. Aufgrund des eingeschränkten Kippwinkels sind die Strukturfaktoren im Bereich des *Missing Wedge* unzugänglich. Von diesem Bereich abgesehen werden die Strukturfaktoren des Objekts bis zur Frequenz des Crowther Kriteriums homogen aufgenommen, die darüber hinausgehende Information ist unvollständig. (In Teilen geänderte Abbildung [Lucic et al., 2005])

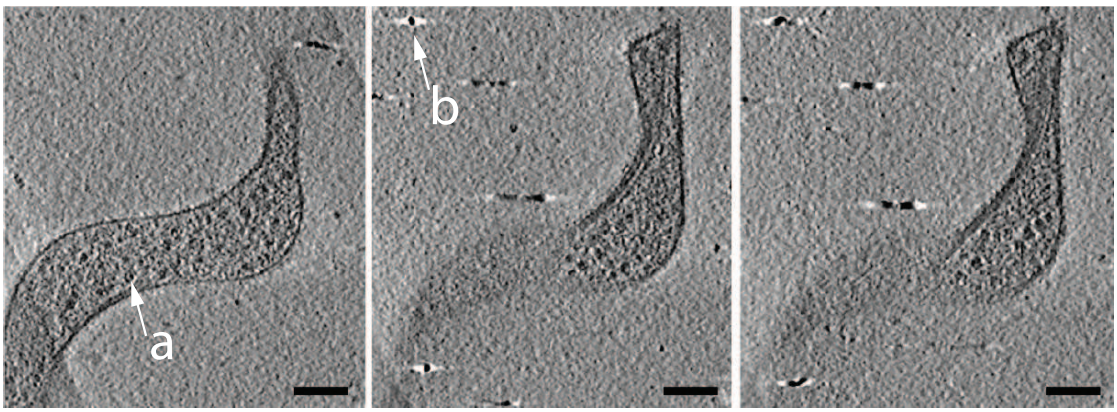


Abbildung 2.3: Orthogonale Schnitte durch eine kryo-elektronentomographische Rekonstruktion des Fadenwurm S. citri. Die Rekonstruktion wurde aus 120 Einzelbildern erstellt. a) Molekulare Strukturen innerhalb des Organismus. b) Goldmarker dienen zur Alignierung der Einzel-Projektionen. Der Balken entspricht 100 nm [Fleischer, 2010]

Eine deutliche Verbesserung der Rekonstruktionsqualität kann durch die Aufnahme einer Zweiachsenkippserie erreicht werden [Nickell, 2001] [Penczek et al., 1995]. Dabei wird zuerst eine Einachsenkippserie des Objekts an der gleichen Probenstelle aufgenommen. Dann wird das Objekt unter Kryobedingung 90° um eine Achse parallel zur optischen Achse des TEM gedreht und eine zweite Einachsenkippserie des Objekts aufgenommen. Bei der Rekonstruktion reduziert sich der *Missing Wedge* auf einen pyramidenförmigen Bereich. Das führt vor allem zu einer geringeren Abhängigkeit der Objektstrukturen von ihrer räumlichen Orientierung.

Die entwickelten Analyseverfahren zur Mustererkennung in den Tomogrammen sind in der Lage, sowohl einen *Missing Wedge*, als auch eine *Missing Pyramid* oder jede andere geometrische Form der Datenlücke zu berücksichtigen. Zur Vereinfachung werden die fehlenden Daten im folgenden als *Missing Wedge* bezeichnet, stellvertretend für alle anderen möglichen Geometrien.

2.4 Mustererkennung

Die Analyse makromolekularer Komplexe innerhalb einer tomographischen Rekonstruktion wird häufig erst durch Verfahren der Mustererkennung möglich. Dabei

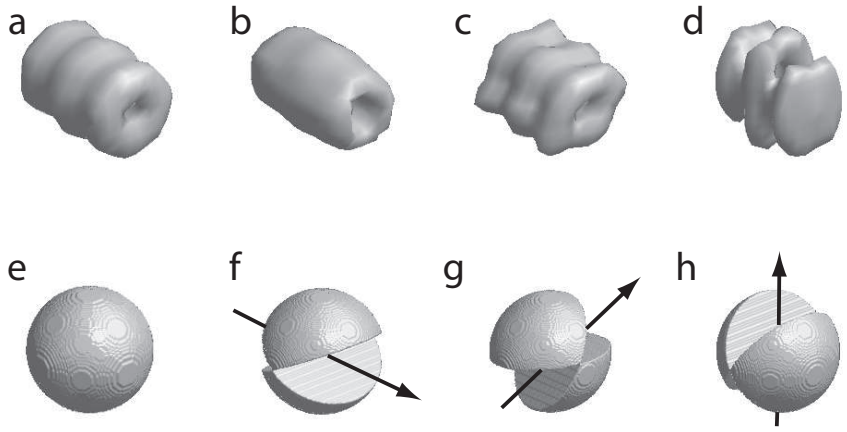


Abbildung 2.4: Einfluß des *Missing Wedge* auf ein rekonstruiertes Partikelbild. a) Oberflächendarstellung der Abbildung eines tiefpassgefilterten 20S Proteasom bei dem e) alle Strukturfaktoren vorhanden sind. b-d) Verzerrungen des Partikels durch eine unterschiedliche Lage zur Kippachse. f-h) Korrespondierender *Missing Wedge* [Stölken, 2008].

gibt es in der Bildverarbeitung eine Vielzahl von möglichen Methoden. Beim klassischen Vorgehen wird die Mustererkennung in zwei Phasen aufgeteilt, die Bildsementierung und die Klassifikation der sementierten Daten [Duda et al., 2001a]. Bei den elektronenmikroskopischen Tomogrammen werden im ersten Schritt interessante Bereiche als Subtomogramm ausgewählt, die z.B. mit hoher Wahrscheinlichkeit molekulare Komplexe enthalten. Im zweiten Schritt werden diese Subtomogramme analysiert, um Klassen mit gleichen Strukturen zu identifizieren (Abb.: 2.5).

Eine häufig verwendete Methode für die Identifikation interessanter Bereiche ist das *Template Matching* [Roseman, 2000] [Frangakis et al., 2002]. Bei diesem Bild-Segmentierungsverfahren werden alle Positonen im Tomogramm (prinzipiell ein aus Grauwerten bestehendes Volumen) mit einer Referenzstruktur verglichen. Sind die gesuchten Strukturen bekannt, können als Referenzen Proteinstrukturen aus der *Protein Data Bank* (PDB) oder anderen Quellen verwendet werden, die durch Filtertechniken der Bildverarbeitung auf die Auflösung des Tomogramm reduziert werden. Der auf Kreuzkorrelation basierende Vergleich der Referenzstruktur mit allen möglichen Positionen und Orientierungen im Tomgramm kann aber auch mit primitiven Strukturen durchgeführt werden, wie z.B. einer Kugel oder einem Zylinder. Die Analyse ergibt ein Korrelationsvolumen, indem die höchsten Korrelationskoeffizienten als Maß für die Übereinstimmung der Referenz mit dem korrespondierenden Bereich in der Probe interpretiert werden. Am Ende der Segmen-

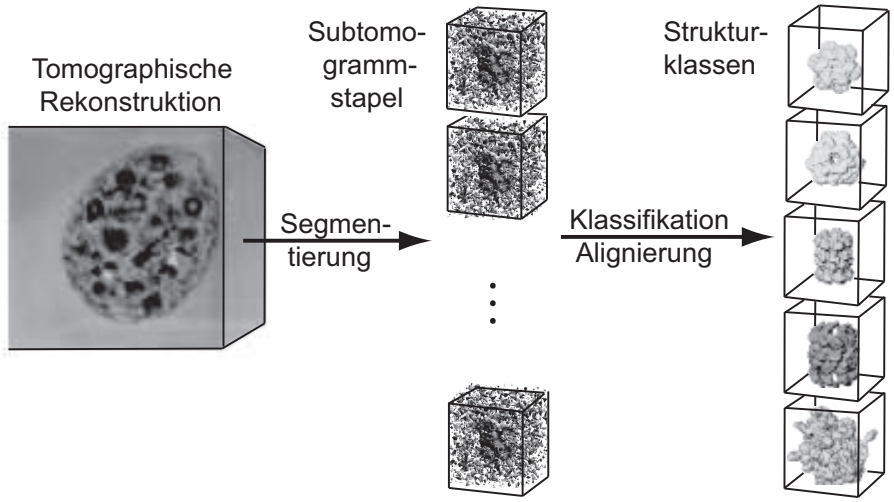


Abbildung 2.5: Mustererkennung. Interessante Punkte in den Tomogrammen werden identifiziert und als Subtomogramme ausgeschnitten. Der Subtomogrammstapel wird mit Klassifikations- und Alignierungsverfahren untersucht, um die Subtomogramme in Strukturklassen aufzuteilen und innerhalb der Klassen zusammen zu mitteln.

tierung werden Subtomogramme an den Positionen der Maxima im Tomogramm ausgeschnitten. Dabei wird die Größe so gewählt, das der größte zu erwartende Komplex vollständig in ein Subvolumen passt.

So entsteht eine Menge oder ein Stapel von Subtomogrammen. Die potenziell enthaltenen 3D-Abbildungen biologischer Strukturen können genauer untersucht werden. Für die Untersuchung des Subtomogrammstapels eignet sich die Anwendung maschineller Lernverfahren. Bayessche Modelle haben sich neben Kernel basierten Verfahren in der Bildverarbeitung zu wichtigen Methoden entwickelt [Duda et al., 2001b] [Schölkopf and Smola, 2002] [Bishop, 2006]. In dieser Arbeit wurden maschinelle Lernverfahren entwickelt und auf das Problem der Klassifikation und Alignierung von Subtomogrammen angewendet. Das Bayessche Entscheidungsmodell war dafür eine wichtige Grundlage.

3 Verfahren zur Analyse von Subtomogrammen

Das Ziel der Analyse von Subtomogrammen ist die Identifikation von Partikelstrukturen in einer Probe und die Optimierung der resultierenden Auflösung der Partikel. Die Analyse kann damit in zwei Teilprobleme aufgetrennt werden: (1) das Alignierungsproblem und (2) das Klassifikationsproblem. Durch die Lösung des Alignierungsproblems ist es möglich, über die alignierten Partikelbilder eine Mittelung zu berechnen und damit das Signal in den Bildern gegenüber dem Rauschen zu verstärken. Je größer die Zahl der Subvolumina ist, die in die Mittelung eingehenden, je stärker wird das Bildsignal, umso besser ist die Rauschunterdrückung und umso höher ist die Auflösung bei einem gegebenen Signal-Rausch-Verhältnis der Ursprungsdaten. Dieser Zusammenhang gilt unter der Annahme, dass alle Subvolumina die selben Partikelstrukturen enthalten und dass der Mittelung eine fehlerfreie Alignierung zu Grunde liegt. Liegen in den Subtomogrammen unterschiedliche Strukturen vor, so ist eine Alignierung und Mittelung über alle Subtomogramme nicht sinnvoll. In diesem Fall ist es zielführend gleiche Partikelstrukturen in Klassen zusammenzufassen. Innerhalb der Klassen können die Strukturen aligniert und gemittelt werden. Schwierig ist die Alignierung und Klassifikation von Subtomogrammen vor allem wegen des initial sehr niedrigen Signal-Rausch-Verhältnisses und des eingeschränkten Kippwinkelbereiches, mit dem daraus resultierenden *Missing Wedge* in der 3D Rekonstruktion.

3.1 Überblick

Eine grundlegende Strategie für die Alignierung von zwei Volumina basiert darauf, eine Kreuzkorrelationsfunktion zwischen zwei Volumina zu berechnen. Jeder Punkt der Funktion entspricht einem Kreuzkorrelationskoeffizienten, der für jede mögliche Translation und Rotation der Volumina zueinander berechnet wird. Der maximale Kreuzkorrelationskoeffizient entspricht der Transformation mit der

besten Übereinstimmung der Strukturen und ist damit das zu optimierende Abstandsmaß bzw. die zu optimierende Metrik.

Die Alignierung kann durch den *Missing Wedge* stark beeinflusst werden. Im Extremfall alignieren die Subtomogramme nicht nach der Orientierung der beinhalteten Partikelstrukturen, sondern auch nach der Lage des *Missing Wedge*. Um dieses Problem zu minimieren, kann die Berechnung der Metrik auf den Bereich der kontinuierlich gefüllten Daten in den Tomogrammen beschränkt werden. Bei der so genannten *Constrained Correlation* wird nur der Bereich außerhalb des *Missing Wedge* für die Berechnung der Metrik berücksichtigt [Frangakis et al., 2002] [Förster, 2005] [Schmid and Booth, 2008] [Bartesaghi et al., 2008]. Für die Berechnung der Metrik, ist u.a. die Drehung der Subvolumina gegeneinander notwendig. Um die rechenintensive vollständige Abtastung der möglichen Rotationen einzuschränken, verwendet Bartesaghi [Bartesaghi et al., 2008] eine Methode, bei der das Faltungstheorem in der Kugelflächenfunktion ausgedrückt wird.

Nach der Alignierung liegen die Alignierungsparameter vor. Sie entsprechen dem minimalen Abstand der Metrik und beschreiben durch eine Rotation und eine Translation für jedes Subvolumen die relative Orientierung der Partikel in der Probe zueinander. Die Summierung der Grauwerte aller alignierten Subtomogramme ergibt ein gemitteltes Partikelbild. Da jedes alignierte Subtomogramm einen individuell orientierten *Missing Wedge* besitzt, muss die resultierende Struktur ggfs. durch einen gemittelten *Missing Wedge* (*compound wedge*) korrigiert werden, um eine gleichmäßige Verteilung der Daten im Fourierraum zu erhalten [Walz et al., 1997].

Die bisherigen Ansätze zur Analyse von Subtomogrammen folgen einem Zweischrittverfahren, bestehend aus einer initialen Alignierung, gefolgt von einer Klassifikation der Subtomogramme. Nach der Alignierung auf eine Referenzstruktur, können die strukturellen Varianzen der Subtomogramme untereinander mittels Klassifikation untersucht werden. Analog zur Alignierung wird auch bei der Klassifikation ein Abstandsmaß verwendet, um die Ähnlichkeit von Subtomogrammen untereinander zu messen. Das Abstandsmaß ist die Basis, auf der Klassifikationalgorithmen die Subtomogramme in unterschiedliche Klassen gruppieren. Es wurde bereits gezeigt, dass die *Constrained Correlation* als Abstandsmass [Förster et al., 2008] bessere Ergebnisse liefert, als die Berechnung der Metrik über einen unbeschränkten Kreuzkorrelationskoeffizienten. Dabei wurde als Klassifikationsalgorithmus hierarchisches *Clustering* oder eine Hauptkomponentenanalyse (PCA) gefolgt vom *k-Means Clustering* eingesetzt.

Bei der Analyse in Kunststoff eingebetteter Spikes der Myosin V und des Simian immunodeficiency Virus Kapsids, berichtete Winkler et al. von verschiedenen Protokollen für die Subtomogramm-Analyse: einem iterativen Alignierungs- und Klassifizierungsprotokoll basierend auf der *Constrained Correlation* Metrik mit einer Multi-Referenz-Analyse und einem referenzfreien Alignierungsschema, bei dem ein Klassifikationsverfahren verwendet wird, um Moleküle mit unterschiedlichen Orientierungen in der Probe in Gruppen zusammenzufassen. Die resultierenden Klassen mit einem verstärkten Signal im Verhältnis zu einzelnen Subtomogrammen, bilden hier die Grundlage für die Alignierung (weitere Details siehe [Winkler et al., 2009]).

Eine generelle Limitierung der genannten Analyseansätze liegt in der Trennung von Alignierung und Klassifikation in zwei aufeinander folgende Schritte. Wenn die Alignierug fehlschlägt, kann im Klassifikationsschritt kein valides Ergebnis erreicht werden. Eine solche Fehlalignierung ist dann wahrscheinlich, wenn sich die Partikel innerhalb eines Datensatzes stark in Struktur und Größe unterscheiden. Für solche Datensätze sind die bisherigen Verfahren oft weniger geeignet, da das Signal-Rausch-Verhältnis für die abgebildeten Strukturen sehr gering ist. Der typische Bereich des Signal-Rausch-Verhältnisses von Kryo-Elektronentomgrammen reicht von 0.01 bis 0.1.

Ein alternativer Ansatz zur Alignierung von 2D Bildern basiert auf der ML-Optimierung und wurde von F.J. Sigworth [Sigworth, 1998] eingeführt und in den folgenden Jahren von Scheres et al. für die 3D Rekonstruktion und Klassifikation für die Einzelpartikel-Analyse erweitert [Scheres et al., 2005]. Der verwendete ML-Algorithmus basiert auf einem Wahrscheinlichkeitsmodell, das nicht nur Informationen über die mögliche Struktur der makromolekularen Komplexe berücksichtigt, sondern auch eine formale Beschreibung des Rauschens und die Verteilung der Alignierungsparameter mit einbezieht. Das Ziel des ML-Algorithmus ist es, die Parameter zu finden, die mit der höchsten Wahrscheinlichkeit die experimentellen Daten unter dem gegebenen Modell beschreiben. Das Modell wird auf eine Wahrscheinlichkeitsfunktion abgebildet, wobei die relative Orientierung und Klassenzuordnung der Bilder als verdeckte Variablen behandelt werden. Um eine Lösung für dieses Problem mit unvollständigen Daten zu finden, kann der *Expectation-Maximizaton*-Algorithmus (EM-Algorithmus) [Dempster et al., 1977] eingesetzt werden. Der EM-Algorithmus gewährleistet durch die Maximierung die Wahrscheinlichkeitsfunktion eine iterative Optimierung der Modellparameter.

Der ML-Algorithmus zeigt auch bei stark verrauschten Bildern ein robustes Verhalten. Die Alignierung und Klassifikation wird auf Basis der Modellparameter

berechnet. Dabei besteht ein wichtiger Modellparameter aus gemittelten Partikelstrukturen, in die alle möglichen Orientierungen und Klassen von Einzelpartikelbildern gewichtet mit einer Wahrscheinlichkeitsdichtefunktion eingehen. Ein solcher Ansatz einer Multi-Referenz Klassifizierungsstrategie für die gleichzeitige Alignierung und Klassifikation von Subtomogrammen wurde von Scheres vorgestellt [Scheres et al., 2009]. In Analogie zum Ansatz der Einzelpartikel-Analyse, werden zur Lösung des Klassifizierungs- und Alignierungsproblems Orientierungen und Klassenzugehörigkeiten als verdeckte Variablen behandelt. Um die fehlenden Daten in den tomographischen Rekonstruktionen zu berücksichtigen, werden die Datenbereiche innerhalb der *Missing Wedge* ebenfalls als verdeckte Variablen behandelt. Die gemittelten Strukturen aller Klassen werden auf Basis der wahrscheinlichkeitsgewichteten Mittelung über alle möglichen Orientierungen und Klassenzugehörigkeiten berechnet. Der Bereich der fehlenden Information ergibt sich direkt aus den Modellparametern. Somit ist keine weitere Buchhaltung über die Orientierung des *Missing Wedge* jedes Subtomogramms notwendig und die Korrektur der resultierenden Klassenmittelungen, die aufgrund der anisotropen Abtastung im Fourierraum entsteht, geschieht implizit.

3.2 Beschreibung des Alignierungsproblems

Formal kann das Alignierungsproblem wie folgt beschrieben werden. Gehen wir davon aus, dass jedes Subtomogramm X_i aus der Menge der untersuchten Subtomogramme $\mathbf{X}$ eine rotierte und translatierte Kopie derselben unbekannten Struktur A ist, deren Signal durch Rauschen überlagert ist. Aufgrund des experimentellen Aufbaus wird jedes Subvolumen von einem *Missing Wedge* W_i beeinflusst, der je nach Kippung des Probenhalters bei der Aufnahme unterschiedliche Öffnungswinkel je Subtomogramm haben kann. Wenn wir das Rauschen als additives Volumen N_i betrachten, können wir die Daten wir folgt beschreiben:

$$R_{\phi_i} X_i = (R_{\phi_i^r} W_i^*) \otimes (A + N_i). \tag{3.1}$$

dabei ist $X_i \in \mathbb{R}^M$ bestehend aus M Voxel, jeder bezeichnet mit einem Index $m \in \{1, 2, ...M\}$ und ist i'te untersuchte Subvolumen $i \in \{1, 2, ...N\}$; A ist die unbekannte, unter den Daten liegende Struktur; $\otimes$ ist der Faltungsoperator; R_{ϕ_i} definiert die Transformation von X_i auf die Position von A; ϕ_i beinhaltet die 3D Rotation ϕ_i^r und die Translation ϕ_i^t beschrieben in kartesischen Koordinaten:

$$\begin{aligned} \phi_i &= (\phi_i^t, \phi_i^r) \\ \phi_i^t &= (b_{xi}, b_{yi}, b_{zi}). \end{aligned} \tag{3.2}$$

$N_i \in \mathbb{R}^M$ ist das additive Rauschen und $W_i^* \in \mathbb{R}^M$ ist die Fouriertransformierte (FT) des *Missing Wedge* in den Realraum:

$$W^* = FT(W) \text{ with } W = \begin{cases} 1, & \text{wenn Voxel im ereichbaren Kippwinkelbereich} \\ 0, & \text{sonst.} \end{cases} \tag{3.3}$$

Nun sind alle X_i und die dazugehörigen W_i bekannt. Unbekannt ist die den Subvolumina X_i zugrunde liegende Struktur A. Ziel ist es, die Rotation und Translation R_{ϕ_i} für jedes Subvolumen X_i zu finden, so dass das Bild in die Position von A transformiert wird bzw. dass alle Subvolumina X_i dieselbe relative Orientierung zueinander besitzen. Ist R_{ϕ_i} für alle i bekannt, so lässt sich eine gemittelte Struktur A^w wie folgt berechnen:

$$A^w = \frac{1}{N} \sum_{i=1}^{N} R_{\phi_i} X_i. \tag{3.4}$$

Für den Fall, dass die Orientierungen der Partikel einer Gleichverteilung folgen, gilt $A^w = A$. Für den allgemeineren Fall, dass die Partikel nicht gleichverteilt orientiert vorliegen, kann A^w wie folgt korrigiert werden:

$$A = \frac{1}{\overline{W}^*} \otimes A^w, \tag{3.5}$$

wobei $\overline{W}^*$ die Fouriertransformierte des *compound Wedge* $\overline{W}$ ist:

$$\overline{W} = \frac{1}{N} \sum_{i=1}^{N} R_{\phi_i^r} W_i. \tag{3.6}$$

3.3 Distanzmaße und -funktionen

Allgemeine Lösungsstrategien, die sich auf das Alignierungsproblem anwenden lassen, bestehen häufig aus zwei wichtigen Komponenten. Im Kern befindet sich eine Metrik. Sie ist ein Distanzmaß, um den Unterschied zwischen zwei Objekten zu messen. Die zweite Komponente ist der Klassifizierer, der auf Basis der Messungen der Metrik eine Entscheidung für die Zugehörigkeit zu Klassen trifft (in unserem Fall ist jede mögliche Orientierung des Objektes eine Klasse). In die Metrik können verschiedene Faktoren einfließen mit dem Ziel, die realen Zusammenhänge so genau wie möglich abzubilden, und dennoch die Komplexität der Metrik so zu wählen, dass die Berechenbarkeit gegeben bleibt.

3.3.1 Der quadratische Euklidische Abstand

Betrachten wir die zwei Volumina $X \in \mathbb{R}^M$ und $A \in \mathbb{R}^M$ und bestehend aus M Voxel und jeder bezeichnet mit einem index $m \in \{1, 2, ...M\}$ (wie bereits in Abschnitt 3.2). Die zu untersuchenden Bilder sind 3D-Volumina. Mit der gewählten Darstellung werden alle Voxel indiziert und als Zeilenvektor behandelt. Jeder Voxel kann einen Grauwert in einem definierten Bereich der reellen Zahlen annehmen. Nun soll eine Aussage getroffen werden, ob und wie stark die Bilder sich unterscheiden. Ein einfaches Maß, um Unterschiede zwischen den Bildern zu messen, ist der quadratische Euklidische Abstand, der die mathematischen Bedingungen einer Metrik erfüllt:

$$\|X - A\|_2 = \sum_{m=1}^{M} (X_m - A_m)^2. \tag{3.7}$$

Die Metrik berechnet die quadrierte Differenz jedes Voxel von X an der Stelle m mit dem korrespondierenden Voxel von A an der Stelle m und summiert die Differenzen. Sind X und A identische Volumina, so sind alle Differenzen Null und das Ergebnis der Messung ist ebenfalls Null. Im Falle unterschiedlicher Bilder ergibt die Metrik einen Wert > 0 und steigt bei einer linearen Abweichung quadratisch an.

Die Metrik ist translations- und rotationsvariant, d.h. sind identische Strukturen in den Subvolumina gegeneinander verdreht oder verschoben, so ergibt sich ein Abstand > 0. Die Eigenschaft lässt sich nutzen, um eine Distanzfunktion zu beschreiben, indem man eines der beiden Volumina gegen das andere translatiert und rotiert und für jede Transformation R_ϕ (3.1,3.2) den Abstand der beiden Volumina zueinander berechnet:

$$D(\phi) = \|R_\phi X - A\|_2. \tag{3.8}$$

Wenn wir davon ausgehen, dass das Abstandsmaß dieser Distanzfunktion den Unterschied zwischen zwei Subvolumina widerspiegelt, so ergibt sich die Transformation für die Alignierung von zwei Partikeln direkt aus dem Minimum der Funktion $D(\phi)$.

Zwei Volumina mit identischen Strukturen in gleicher Orientierung mit einer unterschiedlichen Skalierung bzw. Amplitude würden in der Metrik zu einem Abstand > 0 führen. Die Skalierung der Daten ist in der Bildverarbeitung oft willkürlich und trägt keine relevante Information. Um die Skalen der Volumina anzupassen,

werden alle untersuchten Volumina auf die Standardabweichung eins und den Mittelwert null normiert:

$$V_{norm} = \frac{V - \overline{v}}{\sqrt{\sum_{m=1}^{M}(V_m - \overline{v})^2}.} \tag{3.9}$$

Dabei ist V das nicht normierte Volumen, $\overline{v}$ der Mittelwert des Volumens V und V_{norm} das normierte Volumen.

Unter der Bedingung, dass die Volumina nach (3.9) normiert sind, lässt sich die Distanzfunktion (3.8) über den Kreuzkorrelationskoeffizienten berechnen [Best et al., 2007]:

$$D(\phi) = 2 - 2\ CCF(R_{\phi^r}X, A)\ [R_{\phi^t}]. \tag{3.10}$$

Dabei ist das Resultat der Kreuzkorrelationsfunktion (CCF) ein Volumen, in dem jede Position eines Voxels als Translation von X gegenüber A verstanden werden kann. Der dazugehörige Voxelwert entspricht dem quadratischen Euklidischen Abstand für die so zueinander verschobenen Volumina. In der Gleichung geben die Klammern [] den Funktionswert (Voxelwert) der CCF für eine bestimmte Translation ϕ^t an. Die CCF lässt sich im Fourierraum über die Faltung effizient berechnen. Statt einer Laufzeit von $O(m^2)$ im O-Kalkül [Sipser, 1997], ermöglicht die schnelle Fouriertransformation (*Fast Fourier Transform*) (FFT) [James W. Cooley, 1965] eine Gesamtlaufzeit von $O(mlog(m))$ für die Berechnung der Distanzfunktion (Gl.: 3.8).

3.3.2 *Constrained Correlation*

Der quadratische Euklidische Abstand ist ein einfaches intuitives Abstandsmaß bei dem eine schnelle Berechenbarkeit gegeben ist. Er ist die Grundlage für die weiteren Betrachtungen. Bei tomographischen Subvolumina entsteht durch die Einschränkung des Kippwinkels bei der Aufnahme ein Bereich im Fourierraum (*Missing Wedge*), der nicht abgetastet wird (Kapitel 2). In jedem Subtomogramm fehlt dieser Bereich an Informationen. Dabei ist die absolute Lage des *Missing Wedge* im Tomogramm und in jedem Subtomogramm identisch. Die Lage relativ zur Partikelorientierung ist durch die vielfältigen Orientierungen der Partikel im Eis aber unterschiedlich. Zusätzlich kann der Öffnungswinkel des *Missing Wedge* von einer Kippserie zur nächsten variieren und ist damit für jedes Partikel individuell festgelegt.

Nehmen wir an, die verwendete Referenz entspricht exakt der zugrunde liegenden Struktur des Partikel. Nehmen wir weiter an, Partikel und Referenz sind frei von

Rauschen, sind gleich orientiert und gleich normiert. Der Vergleich zweier solcher Volumina sollte den Abstand Null ergeben. Verwendet man den quadratischen Euklidischen Abstand, so wäre der Abstand nur dann Null, wenn im Partikelbild keine Informationen fehlen. Durch die fehlenden Bereiche bzw. den *Missing Wedge* wird die Metrik einen Abstand größer Null errechnen. Um diesen generellen Unterschied zwischen Partikel und Referenz zu berücksichtigen, wurde von Förster et al. die *Constrained Correlation* Methode entwickelt [Förster et al., 2008]. Im Kern dieser Methode befindet sich eine Metrik, bei der der fehlende Bereich des Partikel auch in der Referenz maskiert wird (Abb: 3.1):

$$\|X_i - A \otimes W_i^*\|_2. \tag{3.11}$$

In der Metrik werden Frequenzbereiche der Referenz A durch die Faltung mit dem inversen Fouriertransformierten des *Missing Wedge* W_i^* des Partikel X_i auf Null gesetzt, die auch im Partikel keine Information tragen.

Mit dieser Beschränkung der Frequenzen der Referenz ergibt die Metrik einen Abstand von Null, wenn die zuvor genannten Bedingungen gelten (Referenz und Partikel haben die gleiche zugrunde liegende Struktur, sind frei von Rauschen, gleich orientiert und gleich normiert).

Wie zuvor gehen wir davon aus, dass die Partikel in der Probe unterschiedlich orientiert sind. Somit müssen Referenz und Partikel unter den möglichen Translationen und Rotationen verglichen werden. Daraus ergibt sich die folgende Distanzfunktion, deren Minimum die optimale Alignierung bzgl. dieser Metrik beschreibt:

$$D(\phi) = \|R_{\phi_i} X_i - A \otimes (R_{\phi_i^r} W_i^*)\|_2. \tag{3.12}$$

Aus der Distanzfunktion wird deutlich, dass das Partikel X_i mit dem Operator R_{ϕ_i} sowohl translatiert als auch rotiert wird. Die fehlenden Frequenzen des Partikels hängen durch die nicht rotationssymmetrische Keilform des *Missing Wedge* von der Rotation, nicht aber von der Translation des Partikels im Subvolumen ab. Aus diesem Grund wird der *Missing Wedge* W mit dem Operator $R_{\phi_i^r}$ in der Distanzfunktion nur rotiert und nicht translatiert.

3.3.3 Die *Compound Wedge* Metrik

Die Metrik der *Constrained Correlation* Methode berücksichtigt die fehlenden Informationen des Partikels und vergleicht nur Frequenzen der Referenz, die auch

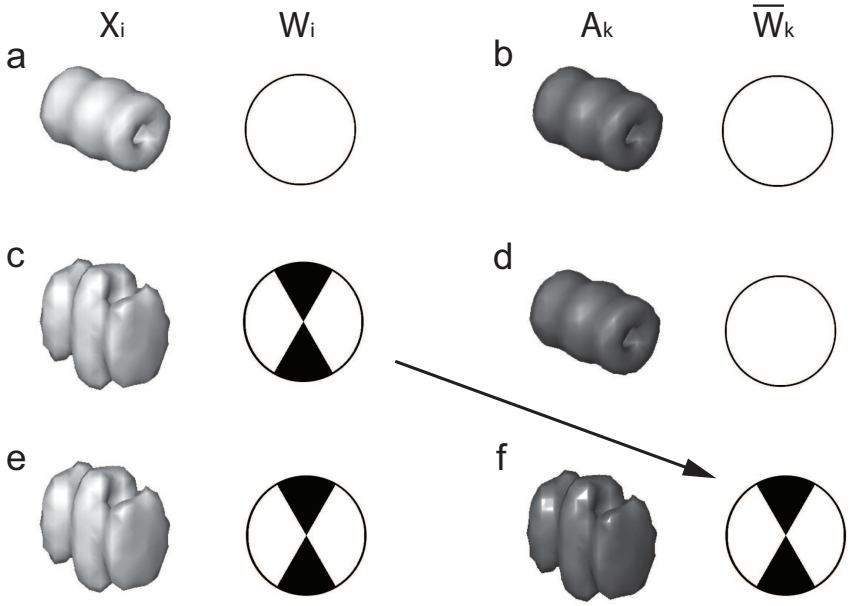


Abbildung 3.1: *Constrained Correlation* Metrik. Kompensation der anisotropen Abtastung des Partikel unter Verwendung der *Constrained Correlation* Metrik. a) Subtomogramm (X_i) mit voller Abtastung der Informationen (links) und dem zentralen Schnitt durch den Fourierraum (rechts). b) Subtomogramm-Mittelung (A^w) mit einer angenommenen isotropen Abtastung. c) Subtomogramm von (a) beeinflusst durch den *Missing Wedge* (W_i) mit 30° Öffnungswinkel. d) Ausgehend von einer isotropen Partikelorientierung ist die Subtomogramm-Mittelung (d) identisch mit (d). e) Das Partikel aus (c) geht unbeeinflusst in die Metrik ein und f) der *Missing Wedge* des Partikels (c) wird mit der Subtomogramm-Mittelung (d) gefaltet ($A \otimes W_i^*$). (e) und (f) geht in eine quadratische Euklidische Norm ein. Die Frequenzbereiche, die im Partikel nicht vorhanden sind, werden in der Referenz maskiert.

im Partikelbild vorhanden sind. Die Metrik geht implizit davon aus, dass die Informationen in der Referenz vollständig und gleichmäßig über alle Frequenzbereiche vorhanden sind. Die in den folgenden Kapiteln beschriebenen iterativen Alignierungs- und Klassifikationsverfahren, ermitteln die Referenz in jeder Iteration auf Basis der berechneten Alignierungs- und Klassifikationsergebnisse aller untersuchten Subromgramme neu. Liegen die Orientierungen der Partikel einer Klasse, die zur neuen Referenz beitragen, gleich verteilt vor, so stimmt die Annahme, dass in der Referenz alle Frequenzen gleichmäßig abgetastet sind. In viele Fällen liegen die Partikel in einer bevorzugten Orientierung im Eis in der Probe vor. Ist dies der Fall, so entstehen in der gemittelten Referenz Frequenzbereiche ähnlich zum *Missing Wedge* des Partikels. Jedoch ist der *Missing Wedge* binär, d.h. entweder die Information ist im Frequenzbereich vorhanden oder nicht. Der *Compound Wedge* der Referenz ist die Summe der alignierten *Missing Wedge*-Volumen der Subtomogramme. Damit kann jeder Strukturfaktor des *Compound Wedge* einen beliebigen diskreten Wert zwischen Null und der Anzahl der Subtomogramme annehmen. Je nach Lage der Partikel ist es möglich, dass die Referenz Frequenzbereiche besitzt, die keine Informationen tragen.

Nun ist die Orientierung, mit der ein Partikel in die Mittelung für eine neue Referenz eingeht, bekannt. Mit (Gl.: 3.6) lässt sich der zusammengesetzte *Missing Wedge* bzw. der *Compound Wedge* der Referenz genau bestimmen. Mit (Gl.: 3.5) kann die Referenz korrigiert werden. In der korrigierten Referenz ist der Frequenzbereich gleichmäßig mit Informationen gefüllt. Verzerrungen im Partikelbild können so korrigiert werden, sofern es keine Frequenzbereiche gibt, in denen keine Informationen gibt.

Bereiche, in denen keine Informationen vorliegen, lassen sich nicht korrigieren. Dies wird auch in (Gl.: 3.5) sofort deutlich, da eine Division durch Null nicht definiert ist. Hinzu kommt, dass bei der Korrektur der Referenz weniger abgetastete Bereiche hoch gewichtet werden. Die Informationen von diesen Bereichen beruhen auf wenigen Messpunkten und haben eine geringere statistische Grundlage. Damit können fehlerhafte Abweichungen in wenigen Partikeln (z.B. entstanden durch Rauschen in den Bildern) eine große Auswirkung auf die korrigierte Mittelung haben. Beide Punkte führen dazu, dass ein Abstandsmaß auf Basis der korrigierten Referenz, instabil werden kann.

Die *Compound Wedge* Metrik wurde entwickelt, um numerische Instabilitäten zu vermeiden. Aus diesem Grund wird in der Metrik die unkorrigierte Referenz A^w (unkorrigierte Mittelung aus den alignierten Subtomogrammen aus der vorherigen Iteration) statt der korrigierten Referenz A verwendet. Analog zum *Missing*

Wedge W_i des Partikels X_i beschreibt der *Compound Wedge* $\overline{W}$ der unkorrigierten Referenz A^w die Anzahl der Messwerte jedes Voxels im Fourierraum.

Gehen wir von dem idealisierten Fall aus, dass Referenz und Partikel die gleiche zugrunde liegende Struktur besitzen, frei von Rauschen sind, gleich orientiert und gleich normiert sind. Die Anzahl der Messwerte je Voxel im Frequenzraum der Referenz ist anisotrop verteilt. In diesem Fall würde auch die zuvor betrachtete *Constraint Correlation* Metrik einen Abstand größer Null anzeigen, obwohl die zugrunde liegende Struktur der Partikel identisch ist. In der *Compound Wedge* Metrik wird zusätzlich (im Vergleich zur *Constraint Correlation* Metrik) die inverse Fouriertransformierte des *Compound Wedge* der Referenz auf das Partikel gefaltet:

$$\|X_i \otimes \overline{W}^* - A^w \otimes W_i^*\|_2. \tag{3.13}$$

Mit dieser Metrik ergibt sich für das gewählte Beispiel ein Abstand und von Null (Abb.: 3.2).

Wie zuvor müssen Referenz und Partikel unter den möglichen Translationen und Rotationen verglichen werden. Daraus ergibt sich für die *Compound Wedge* Metrik die folgende Distanzfunktion, deren Minimum die optimale Alignierung zwischen Partikel und Referenz bzgl. der Metrik beschreibt:

$$D(\phi) = \|(R_{\phi_i} X_i) \otimes \overline{W}^* - A^w \otimes (R_{\phi_i^r} W_i^*)\|_2. \tag{3.14}$$

3.4 Subtomogramm-Alignierung und -Mittelung

Das Alignierungsproblem wurde in Abschnitt 3.2 beschrieben. Wir definieren nun einen Algorithmus, dessen Abstandsmaß auf der vorgestellten *Compound Wedge* Metrik (Gl.: 3.13) beruht. Ziel des Algorithmus ist es, jedem Partikel X_i eine Transformation ϕ_i bestehend aus der Translation ϕ_i^t und der Rotation ϕ_i^r) zuzuordnen, die das Partikel auf die Referenz aligniert. Weiterhin wird durch Mittelung der alignierten Partikel das Rauschen unterdrückt und dadurch die Auflösung der Struktur verbessert.

Gegeben sei die Menge der zu untersuchenden Subtomogramme $\mathbf{X}$ mit den einzelnen Subtomogrammen X_i und deren individuellen *Missing Wedge* W_i. Wir gehen weiter davon aus, dass alle Subtomogramme Bilder von der gleichen darunterliegenden Partikelstruktur enthalten. Wir verwenden eine Referenz A^w und dessen

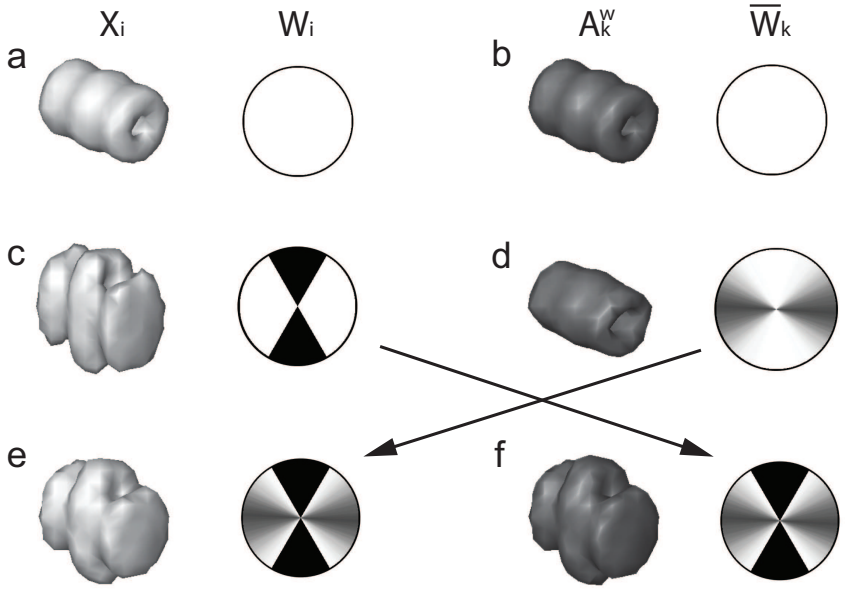


Abbildung 3.2: *Compound Wedge* Metrik. Kompensation der anisotropen Abtastung von Partikel und Referenz unter Verwendung der *Compound Wedge* Metrik. a) Subtomogramm (X_i) mit voller Abtastung der Informationen (links) und dem zentralen Schnitt durch den Fourierraum (rechts). b) Subtomogramm-Mittelung (A^w) mit einer angenommenen isotropen Abtastung. c) Subtomogramm von (a) beeinflusst durch den *Missing Wedge* (W_i) mit 30° Öffnungswinkel. d) Aufgrund von anisotropen Partikelorientierungen ist die Subtomogramm-Mittelung durch den *Compound Wedge* $\overline{W}$ beeinflusst. e) Der *Compound Wedge* von (d) wird mit dem Partikel (c) gefaltet ($X_i \otimes \overline{W}^*$) und f) der *Missing Wedge* des Partikels (c) wird mit der Subtomogramm-Mittelung (d) gefaltet ($A^w \otimes W_i^*$). Beim Vergleichen von (e) mit (f) in einer quadratischen Euklidischen Norm werden nur noch gegenseitig überlappende Frequenzbereiche verglichen.

Compound Wedge $\overline{W}$. Referenz und Partikelbilder sind auf die Standardabweichung eins und den Mittelwert Null normiert (Gl.: 3.9).

Innerhalb einer Iteration wird für ein Subtomogramm X_i mit der Referenz A^w das Minimum der Distanzfunktion (Gl.: 3.14) bestimmt. Um das Minimum zu berechnen, wird für jedes Subtomogramm nach einem vorgegebenen Schemata rotiert. Für jede Rotation lässt sich die *Compound Wedge* Metrik, die eine quadratische Euklidische Norm ist, bzgl. jeder Translation über die CCF im Fourierraum effizient berechnen (Abschitt 3.3.1). Dabei entspricht das Maximum der CCF dem minimalen Abstand. Der Algorithmus bestimmt nach und nach den minimalen Abstand für alle Subtomogramme X_i mit der Referenz A^w. Aus der minimalen Distanz ergeben sich direkt die Translation und Rotation, die das Partikel auf die Referenz aligniert.

Am Ende der Iteration werden die alignierten Partikel gemittelt und ergeben die neue Referenz A^w (Gl.: 3.4) und der zur Referenz gehörende *Compound Wedge* $\overline{W}$ wird ermittelt (Gl.: 3.6). Die neue Referenz und der berechnete *Compound Wedge* sind zusammen mit dem Partikelstapel die Eingangsparameter für die nächste Iteration. Der Algorithmus wird so lange ausgeführt, bis sich die Referenz in der Iteration $t-1$ nicht mehr von der Iteration t unterscheidet oder eine definierte Anzahl an Iterationen erreicht wurde. Zum Abschluss der Alignierung wird die Referenz A_w durch den *Compound Wedge* korrigiert (Gl.: 3.5) und man erhält die korrigierte Mittelung aller Subtomogramme des Stapels (Abb.: 3.3).

Zum Start des Algorithmus muss eine initiale Referenz vorgegeben werden. Je größer die Übereinstimmung zwischen Referenzbild und der wirklichen Partikelstruktur ist, um so kleiner ist die Wahrscheinlichkeit, dass der Algorithmus eine falsche Alignierung und damit eine falsche Mittelung erzeugt. Die initiale Referenz hat in diesem Verfahren einen starken Einfluss auf das Ergebnis. Insbesondere bei stark verrauschten Subtomogrammen, kann die Mittelung ein Bild erzeugen, das nicht die wirkliche in den Daten enthaltene molekulare Struktur widerspiegelt. Der *Compund Wedge* ist zu Beginn der Alignierung normalerweise nicht bekannt. Häufig wird eine isotrop verteilte Orientierung der Partikel angenommen und somit für den initialen *Compund Wedge* ein Subvolumen gewählt, in dem alle Frequenzen gleich gewichtet sind. Bereits in der zweiten Iteration setzen sich Referenz und *Compound Wedge* ausschließlich aus der Mittelung der untersuchten Subtomogramme zusammen bzw. aus der Mittelung der verschiedenen *Missing Wegde*.

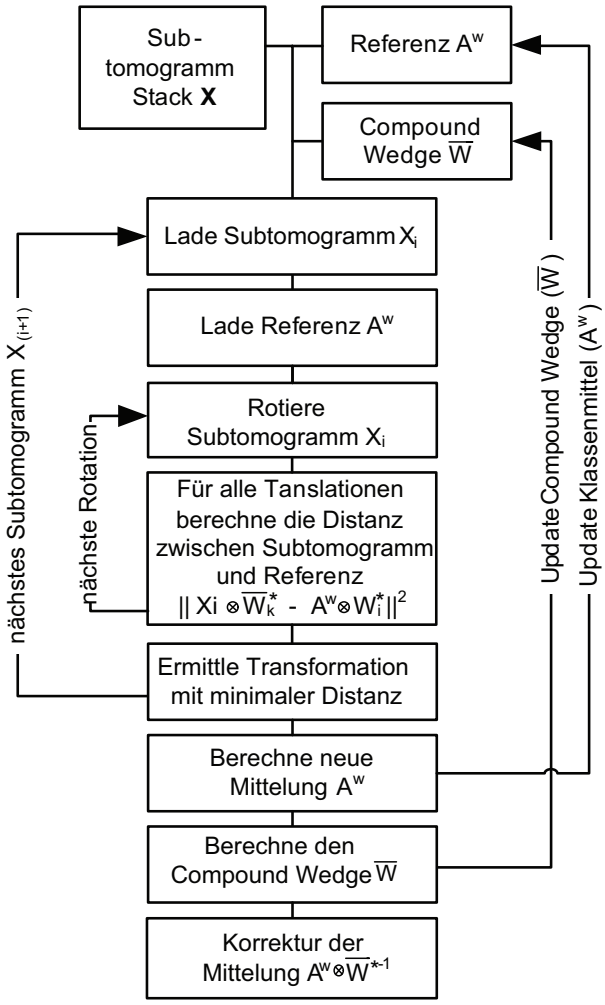


Abbildung 3.3: Algorithmus für die Subtomogramm-Alignierung

3.5 Multi-Referenz-Klassifikation

In den bisherigen Betrachtungen sind wir davon ausgegangen, dass alle Subtomogramme eines Partikelstapels Abbildungen derselben Struktur enthalten. In Tomogrammen von Zellen oder Geweben gibt es eine große Anzahl unterschiedlicher Partikel. Auch in aufgereinigten Proben, in denen nur ein bestimmtes Partikel isoliert wurde, ist es sehr wahrscheinlich, dass ein und dasselbe Partikel in unterschiedlichen Konformationen vorliegt.

Eine Strategie zur Klassifikation der Partikel wird benötigt, um unterschiedliche Partikel bzw. Partikelkonformationen in einzelne Klassen aufzuteilen und die Partikel innerhalb der Klasse zu alignieren und zu mitteln. Hier wählen wir den Ansatz einer Multi-Referenz-Klassifikation und führen multiple Referenzen A_k^w ein, mit $k \in \{1...K\}$ als Index für die k-te Referenz und K als Anzahl der verwendeten Referenzen. Für die Identifikation der richtigen Klasse eines Partikels in einem Subtomogramm wird jedes Partikel statt nur mit einer Referenz mit vielen verschiedenen Referenzen verglichen. Der Abstand eines Subtomogramm X_i wird für jede Position des Partikel für alle verwendeten Referenzen A_k^w berechnet:

$$D(k, \phi) = \|(R_{\phi_i} X_i) \otimes \overline{W}_k^* - A_k^w \otimes (R_{\phi_i^r} W_i^*)\|_2. \tag{3.15}$$

Bisher war die Distanzfunktion (Gl.: 3.14) nur abhängig von der Rotation und Translation ϕ des Subtomogramms X_i gegenüber einer Referenz. Die Distanzfunktion (Gl.: 3.15) ist nun zusätzlich abhängig von der Klasse k.

Der Algorithmus zur Multi-Referenz-Klassifikation (Abb.: 3.4) ist eine Erweiterung des Algorithmus zur Alignierung der Partikel aus dem vorherigen Abschnitt. Der Algorithmus iteriert neben den Subtomogrammen X_i und den Rotationen zusätzlich über alle K Referenzen. Er ermittelt die Referenz (bzw. Klasse) und Transformation mit dem minimalen Abstand für jedes X_i. Jedes Subtomogramm wird dabei genau einer Transformation und einer Klasse zugeordnet (Abb.: 3.5). Innerhalb der Klasse werden die Partikel gemittelt und gehen als neue Referenz in die nächste Iteration des Algorithmus ein. Die Anzahl der initialen Referenzen und damit die Anzahl der maximal möglichen Klassen wird vor Beginn der Ausführung des Algorithmus fest vorgegeben und bleibt während der Iterationen konstant. K geht als Faktor in die Laufzeit des Algorithmus ein. Das Problem, dass die initialen Referenzen einen starken Einfluss auf das Ergebnis haben, ist bei der Multi-Referenz-Klassifikation ebenso gegeben, wie bei der Subtomogramm-Alignierung und -Mittelung.

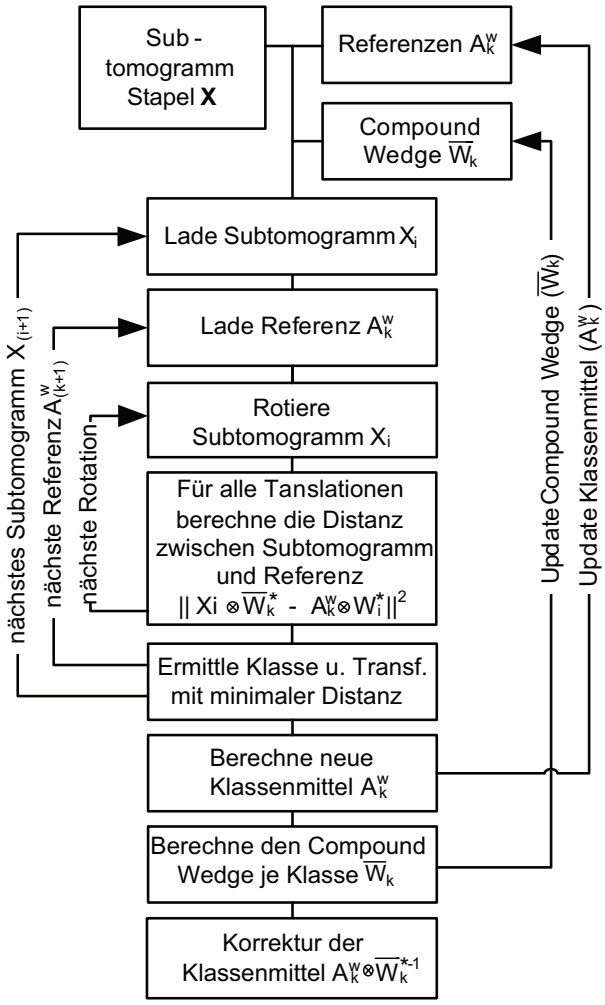


Abbildung 3.4: Algorithmus für die Multi-Referenz-Klassifikation

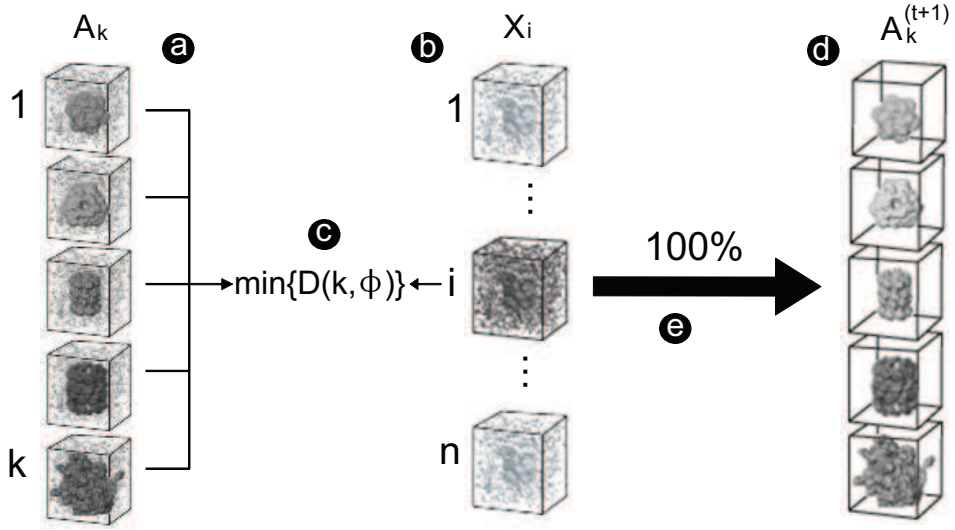


Abbildung 3.5: Multi-Referenz-Klassifikation. a) Referenzen A_k^w legen K verschiedene Klassen fest. b) Subtomogrammstapel X_i mit N Subtomogrammen. c) Referenzen A_k^w und Subtomogramme X_i werden mit der Distanzfunktion (3.15) verglichen und der minimale Abstand wird berechnet. d) Mittelung der Partikel innerhalb der Klassen ergeben die Referenz $A_k^{w\ (t+1)}$ für die nächste Iteration. e) Jedes Partikel X_i fließt ausschließlich in die Mittelung der Klasse mit dem minimalen Abstand ein.

3.6 *Maximum Likelihood* basierte Klassifikation

Die vorgestellte Multi-Referenz-Klassifikation ist in der Lage, in einem iterativen Prozess Subtomogramme gleichzeitig zu alignieren und zu klassifizieren. In jeder Iteration wird jedem Subvolumen genau eine Orientierung und eine Klasse zugewiesen. Die Mittelungen der Partikel einer Klasse ergeben die Startwerte für die darauf folgende Iteration. Es wird sich zeigen, dass diese Art der Optimierung häufig nach wenigen Iterationen zu einem Ergebnis führt, die initialen Strukturen bzw. Referenzen jeder Klasse aber einen starken Einfluss auf das Ergebnis haben. Dies ist insbesondere dann unerwünscht, wenn nicht bekannt ist, welche Strukturen in der Probe enthalten sind.

Der ML-Algorithmus hat das Ziel, den Einfluss der initialen Referenz auf das Ergebnis zu minimieren und kann als Erweiterung der Multi-Referenz-Klassifikation betrachtet werden. Im Kern können dieselben Metriken verwendet werden. Der wesentliche Unterschied liegt darin, dass die Metriken die Basis für Wahrscheinlichkeitsdichtefunktion sind. So wird anstatt einer optimalen Alignierung und Klassifikation für jedes Partikelbild eine gewichtete Summe über alle möglichen Orientierungen und Klassen jedes Partikelbildes berechnet (Abb.: 3.6). Auf Basis der Metrik wird eine Wahrscheinlichkeitsdichteverteilung berechnet, die jeder möglichen Klassifikation eine Wahrscheinlichkeit zuordnet. Das bedeutet, dass die Zuordnung von Subtomogrammen zu einer Partikelklasse und einer Transformation im ML-Verfahren nicht eindeutig ist (Abb. 3.6). Erst im Konvergenzfall des ML-Algorithmus, wenn die Wahrscheinlichkeitsdichtefunktionen in einem Punkt den maximalen Wert erreichen, ist für jedes Partikel eine eindeutige Klasse und Transformation festgelegt. Dann sind die Ergebnisse der ML-Klassifikation und der Multi-Referenz-Klassifikation identisch, vorausgesetzt, es werden dieselben Referenzen und dieselbe Metrik verwendet [Penczek et al., 1992]. Der aufwendigere ML-Optimierungspfad zeigt eine höhere Stabilität zum vorherigen Verfahren, wenn z.B. die initiale Referenz nicht bekannt oder falsch ist. Die Unterschiede der Verfahren werden im Ergebnisteil der Arbeit untersucht.

3.6.1 Die *Likelihood*-Funktion

Ziel des ML-Optimierungsverfahren ist es, wie auch bei der Multi-Referenz-Klassifikation, alle Strukturen A_k in Form von Klassenmittelungen in der Menge von Subvolumen $\mathbf{X}$ zu identifizieren. Grundlage für die Anwendung der ML-Methode ist die Parametrisierung der *Likelihood* Funktion. Da für jedes Subtomogramm X_i

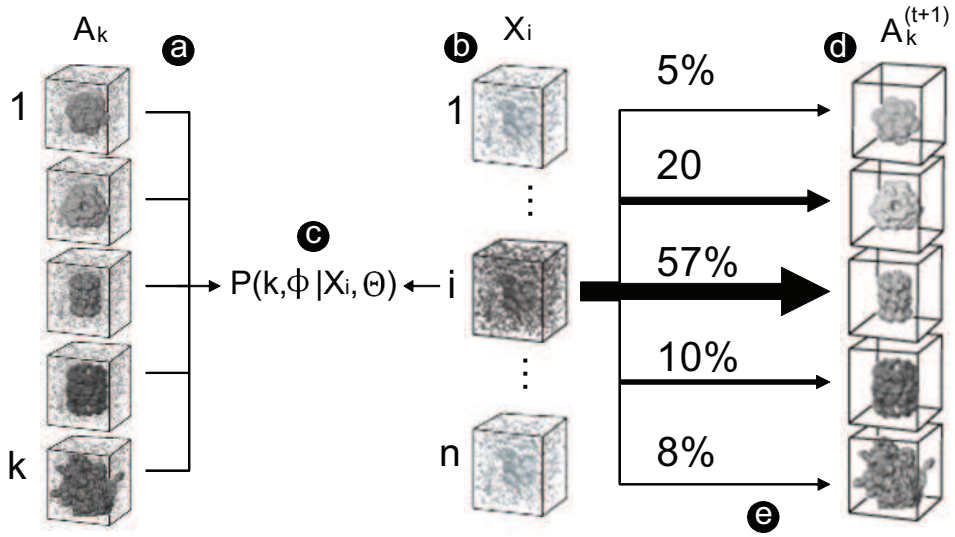


Abbildung 3.6: *Maximum Likelihood* Klassifikation. a) Referenzen A_k^w legen K verschiedene Klassen fest. b) Subtomogrammstapel X_i mit N Subtomogrammen. c) Die Posteriore Wahrscheinlichkeit (3.25) wird für jedes Partikel X_i unter Betrachtung aller K Referenzen A_k^w berechnet. Sie beschreibt (neben einer Wahrscheinlichkeit für jede mögliche Transformation der Subtomogramme bzgl. der Alignierung zu einer Referenz) die Zugehörigkeit der Subtomogramme X_i zu den Klassen. d) Die Neuberechnung der Modellparameter ergibt u.a. die Referenzen $A_k^{w\ (t+1)}$ für die nächste Iteration. e) Entsprechend der Wahrscheinlichkeitsdichteverteilung (Posteriore Wahrscheinlichkeit) kann jedes Partikel X_i zu unterschiedlichen Anteilen in die Mittelung mehrerer Klassen eingehen.

die Transformation ϕ_i und die Klassenzugehörigkeit k_i unbekannt ist, behandeln wir ϕ_i und k_i als verdeckte Variablen. Allgemein maximiert das Verfahren eine Gesamtwahrscheinlichkeit $P(X|\Theta)$ bzw. die Wahrscheinlichkeit der Beobachtungen $\mathbf{X}$ gegebenen einem Modell mit den Parametern Θ. Es beinhaltet den Parameter σ für das angenommene Gaußsche Rauschen; den Parameter $a_{k\phi^r}$, der die A-priori-Wahrscheinlichkeit, dass eine bestimmte Klasse oder Rotation vorliegt, modifiziert; den Parameter ξ, der die A-priori-Wahrscheinlichkeit für die Translation der Struktur weg vom Zentrum beeinflusst und die Abschätzung für die unkorrigierten Strukturen A_k^w. Das Maximieren der *Likelihood* Funktion ist äquivalent zur Maximierung seines Logarithmus und kann wie folgt geschrieben werden:

$$\begin{aligned} L(\Theta) &= \sum_{i=1}^{N} log\ P(X_i|\Theta) \\ &= \sum_{i=1}^{N} log \sum_{k=1}^{K} \int P(k,\phi|\Theta) P(X_i|k,\phi,\Theta) d\phi. \end{aligned} \tag{3.16}$$

Dabei ist $P(k,\phi|\Theta)$ die A-priori Wahrscheinlichkeit von k und ϕ gegeben die Modellparameter Θ und $P(X_i|k,\phi,\Theta)$ ist bedingte Wahrscheinlichkeit eines Subvolumen X_i gegeben die k, ϕ, Θ. Im Kern des Algorithmus befindet sich die Metrik, die in die bedingte Wahrscheinlichkeit $P(X_i|k,\phi,\Theta)$ eingebettet ist. Wir verwenden hier die neue *Compound Wedge* Metrik, die im vorherigen Kapitel bereits vorgestellt wurde und hier um einen normalisierenden Faktor erweitert wird. Die Metrik berechnet den Abstand zwischen X_i unter der Transformation ϕ und A_k^w wie folgt:

$$\frac{1}{C_{i\phi_i^r k}} \| (R_{\phi_i} X_i) \otimes \overline{W}_k^* - A_k \otimes (R_{\phi_i^r} W_i^*) \|_2. \tag{3.17}$$

Dabei ist der normalisierende Term $C_{i\phi_i^r k} = \sum_{j=1}^{J} \overline{W}_{kj} (R_{\phi_i^r} W_i)_j$ und $j \in \{1,2,...J\}$ ist die j'te Komponente im Fourierraum und $\|.\|_2$ bezeichnet die quadratische Euklidische Norm.

Der normalisierende Faktor wird bei der Einbettung der *Compound Wedge* Metrik in das ML-Verfahren gegenüber der Multi-Referenz-Klassifikation notwendig. In der Multi-Referenz-Klassifikation (Absch.: 3.5) wird lediglich der minimale Abstand für eine Alignierungs- und Klassifikationsentscheidung herangezogen. Beim ML-Verfahren wird die gesamte Distanzfunktion ausgewertet. Entsprechend der Überdeckung des Partikels *Wedge* W_i und des *Compound Wedge* $\overline{W_k}$, verändert sich die Anzahl der nicht maskierten Voxel in der Metrik in Abhängigkeit von der Rotation $R(\phi_i^r)$. Damit würde die *Compound Wedge* Metrik zu einer Sensitivität in Abhängigkeit der relativen Orientierung von X_i zu A_k^w kommen. Der normalisierende Term korrigiert mit $\frac{1}{C_{i\phi_i^r k}}$ diesen Effekt in der ML-Methode.

Nach dem Parsevalschen Theorem ist die Metrik (3.17) äquivalent zu seiner Fouriertransformierten:

$$\frac{1}{C_{i\phi_i^r k}} \sum_{j=1}^{J} \left((R_{\phi_i} X_i^*)_j \overline{W}_{kj} - A_{kj}^{w*} (R_{\phi_i^r} W_i)_j \right)^2. \tag{3.18}$$

Dabei ist $j \in \{1, 2...J\}$ die j'te Komponente im Fourierraum.

Wir gehen von einer Überlagerung der Strukturbilder in den Subtomogrammen mit Gaußschem Rauschen aus. Damit folgt der Fehler im Abstandsmaß (3.18) einer Gauß-Verteilung mit der Standardabweichung σ. Wir nehmen weiterhin die Unabhängigkeit für alle Fourier-Komponenten j an:

$$P(X_i|k, \phi, \Theta) = \left(\frac{1}{\sqrt{2\pi}\sigma} \right)^J \exp \left(- \frac{\sum_{j=1}^{J} \left((R_{\phi_i} X_i^*)_j \overline{W}_{kj} - A_{kj}^{w*} (R_{\phi_i^r} W_i)_j \right)^2}{2 C_{i\phi_i^r k} \sigma^2} \right). \tag{3.19}$$

Die A-Priori Wahrscheinlichkeit teilt sich in zwei verbundene Wahrscheinlichkeiten auf:

$$P(k, \phi|\Theta) = P(k, \phi^r, \phi^t|\Theta) = P(k, \phi^r|\Theta) P(\phi^t|k, \phi^r, \Theta). \tag{3.20}$$

Dabei ist $P(k, \phi^r|\Theta)$ die A-Priori Wahrscheinlichkeit, dass ein Partikel zu einer bestimmten Rotation und Klasse gehört:

$$P(k, \phi^r|\Theta) = \alpha_{k,\phi^r}, \tag{3.21}$$

mit $\alpha_{k,\phi^r} \geqslant 0$ und $\sum_k \int_{\phi^r} \alpha_{k,\phi^r} \, d\phi^r = 1$. Wir gehen davon aus, dass durch *Template Matching* (Kapitel 2.4) oder manuelle Selektion die interessanten Partikelstrukturen annähernd zentriert im Subvolumen liegen. Dieser Sachverhalt wird durch eine Wahrscheinlichkeitsdichte über die Translation ϕ^t modelliert. Die Dichtefunktion wird durch eine 3D Gauß-Verteilung zentriert auf den Mittelpunkt des Subvolumen beschrieben:

$$P(\phi^t|k, \phi^r, \Theta) = \frac{1}{(2\pi)^{\frac{3}{2}} \xi^3} \exp \left(- \frac{d_x^2 + d_y^2 + d_z^2}{2\xi^2} \right), \tag{3.22}$$

mit der Standardabweichung ξ, die die durchschnittliche Abweichung der Position der Partikel vom Zentrum über den gesamten Datensatz beschreibt.

Innerhalb der *Likelihood* Optimierung wird der *Compound Wedge* $\overline{W}_k$ als konstanter Wert angenommen. Die folgenden Modellparameter leiten sich aus den Formeln (3.19),(3.21) und (3.22) ab:

$$\Theta = (A_k^w, \sigma, \alpha_{k,\phi^r}, \xi). \tag{3.23}$$

Das Ziel der Optimierung ist es, die Modellparameter zu finden, die den in (3.16) definierten Log-*Likelihood* maximieren.

3.6.2 Der *Maximum Likelihood* Algorithmus

Wir verwenden den *Expectation-Maximizaton*-Algorithmus (EM-Algorithmus) [Dempster et al., 1977] um die *Likelihood* Funktion (3.16) zu maximieren. Der E-Schritt dieses iterativen Algorithmus berechnet den Erwartungswert für die gesamte Log-*Likelihood*-Funktion in Bezug auf die verdeckten Variablen k und ϕ und gegeben alle beobachteten Subvolumen X_i. Dieser entspricht der unteren Grenze $Z(\Theta;\Theta^{(t)})$ für die aktuell gültigen Modellparameter $\Theta^{(t)}$ (t nummeriert die Iterationen):

$$Z(\Theta;\Theta^{(t)}) = \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\ log\{\ P(X_i|k,\phi,\Theta)\ P(k,\phi|\Theta)\ \}d\phi. \quad (3.24)$$

Mit dem Bayestheorem [Bayes, 1991] lässt sich die Posteriore Wahrscheinlichkeit für die Werte von k und ϕ, für die beobachteten Subvolumen X_i und gegebenen Modellparametern $\Theta^{(t)}$ wie folgt berechnen:

$$P(k,\phi|X_i,\Theta^{(t)}) = \frac{P(X_i|k,\phi,\Theta^{(t)})P(k,\phi|\Theta^{(t)})}{\int P(X_i|k',\phi',\Theta^{(t)})P(k',\phi'|\Theta^{(t)})d\phi'}\ . \quad (3.25)$$

Der darauf folgende M-Schritt maximiert den Erwartungswert des E-Schritts in Bezug auf alle Modellparameter. Basierend auf der aktuellen Abschätzung der Modellparameter aus den t'ten Iteration $\Theta^{(t)}$ erhöht der Algorithmus die Wahrscheinlichkeit *Likelihood* für die neuen Modellparameter $\Theta^{(t+1)}$. Dafür setzen wir die partiellen Ableitungen der unteren Grenze (3.24) der *Likelihood* Funktion in Bezug auf jeden Modellparameter gleich Null und lösen die Gleichung nach der gewählten Variablen auf.

Modellparameter $\mathbf{A_k^{w*}}$. Wir berechnen die partielle Ableitung für den Modellparameter A_k^{w*}. Dafür setzen wir (3.19) in (3.24) ein und erhalten:

$$\begin{aligned} & Z(\Theta;\Theta^{(t)}) \\ &= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\Bigg(log\left((\sqrt{2\pi}\sigma)^{-J}\right) \\ &\quad -\frac{\sum_{j=1}^{J}\left((R_{\phi_i}X_i^*)_j\overline{W}_{kj} - A_{kj}^{w*}(R_{\phi_i^r}W_i)_j\right)^2}{2C_{i\phi_i^r k}\sigma^2} + log\left(P(k,\phi|\Theta)\right)\Bigg)\ d\phi. \end{aligned}$$

Die Terme $log\left((\sqrt{2\pi}\sigma)^{-J}\right)$ und $log\left(P(k,\phi|\Theta)\right)$ sind von A_{kj}^{w*} unabhängig (Gl. 3.20) und die partielle Ableitung nach A_{kj}^{w*} ergibt damit:

$$\frac{\partial Z}{\partial A_{kj}^{w*}} = 0$$
$$= \sum_{i=1}^{N} \int P(k,\phi|X_i,\Theta^{(t)}) \left(\frac{2(R_{\phi_i} X_i^*)_j \overline{W}_{kj} (R_{\phi_i^r} W_i)_j - 2A_{kj}^{w*} \left((R_{\phi_i^r} W_i)_j\right)^2}{2C_{i\phi_i^r k}\sigma^2} \right) d\phi.$$

Wir multiplizieren beide Seiten mit $2C_{i\phi_i^r k}\sigma^2$ und unter Berücksichtigung von

$\left((R_{\phi_i^r} W_i)_j\right)^2 = (R_{\phi_i^r} W_i)_j (R_{\phi_i^r} W_i)_j = (R_{\phi_i^r} W_i)_j$ und

$(R_{\phi_i} X_i^*)_j (R_{\phi_i^r} W_i)_j = (R_{\phi_i} X_i^*)_j$

erhalten wir:

$$= \sum_{i=1}^{N} \int P(k,\phi|X_i,\Theta^{(t)}) \left((R_{\phi_i} X_i^*)_j \overline{W}_{kj} - A_{kj}^{w*} (R_{\phi_i^r} W_i)_j \right) d\phi.$$

Nun lösen wir die Gleichung nach A_{kj}^{w*} auf, verallgemeinern für alle j und erhalten den Modellparameter $A_k^{w*\,(t+1)}$ für den E-Schritt in der folgenden Iteration:

$$A_k^{w*\,(t+1)} = \frac{\sum_{i=1}^{N} \int P(k,\phi|X_i,\Theta^{(t)}) \; (R_\phi X_i^*) \; d\phi}{\sum_{i=1}^{N} \int P(k,\phi|X_i,\Theta^{(t)}) \; (R_{\phi^r} W_i) \; d\phi} \; \overline{W}_k. \tag{3.26}$$

Modellparameter σ. Wir berechnen die partielle Ableitung für den Modellparameter σ. Dafür setzen wir erneut (3.19) in (3.24) ein und erhalten:

$$Z(\Theta;\Theta^{(t)})$$
$$= \sum_{i=1}^{N} \sum_{k=1}^{K} \int P(k,\phi|X_i,\Theta^{(t)}) \Bigg(- J \; log(\sqrt{2\pi}\sigma)$$
$$- \frac{\sum_{j=1}^{J} \left((R_{\phi_i} X_i^*)_j \overline{W}_{kj} - A_{kj}^{w*} (R_{\phi_i^r} W_i)_j\right)^2}{2C_{i\phi_i^r k}\sigma^2} + log\left(P(k,\phi|\Theta)\right) \Bigg) d\phi.$$

Der Term $log\left(P(k,\phi|\Theta)\right)$ ist von σ unabhängig (Gl.: 3.20) und die partielle Ableitung nach σ ergibt sich damit zu:

$$
\begin{aligned}
&\frac{\partial Z}{\partial \sigma} = 0 \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)}) \\
&\quad \left(\frac{-J}{\sqrt{2\pi}\sigma}\sqrt{2\pi} + \frac{2\sum_{j=1}^{J}\left((R_{\phi_i}X_i^*)_j\overline{W}_{kj} - A_{kj}^{w*}(R_{\phi_i^r}W_i)_j\right)^2}{2C_{i\phi_i^r k}\sigma^3}\right) d\phi \\
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)}) \\
&\quad \left(-\frac{J}{\sigma} + \frac{\sum_{j=1}^{J}\left((R_{\phi_i}X_i^*)_j\overline{W}_{kj} - A_{kj}^{w*}(R_{\phi_i^r}W_i)_j\right)^2}{C_{i\phi_i^r k}\sigma^3}\right) d\phi.
\end{aligned}
$$

Wir multiplizieren beide Seiten mit σ^3

$$
\begin{aligned}
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)}) \\
&\quad \left(-J\sigma^2 + \frac{\sum_{j=1}^{J}\left((R_{\phi_i}X_i^*)_j\overline{W}_{kj} - A_{kj}^{w*}(R_{\phi_i^r}W_i)_j\right)^2}{C_{i\phi_i^r k}}\right) d\phi.
\end{aligned}
$$

$-J\sigma^2$ ist ein konstanter Faktor und kann vor das Integral und die Summen gezogen werden. Durch Ausmultiplizieren der Klammer erhalten wir:

$$
\begin{aligned}
&= -\sigma^2 J\sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\, d\phi \; + \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)}) \\
&\quad \frac{1}{C_{i\phi_i^r k}}\sum_{j=1}^{J}\left((R_{\phi_i}X_i^*)_j\overline{W}_{kj} - A_{kj}^{w*}(R_{\phi_i^r}W_i)_j\right)^2 \, d\phi.
\end{aligned}
$$

Wir verwenden

$$\sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\, d\phi = N = J$$

und lösen die Gleichung nach σ auf:

$$\sigma^{(t+1)} = \left(\frac{1}{2N} \sum_{i=1}^{N} \sum_{k=1}^{K} \int P(k, \phi | X_i, \Theta^{(t)}) \right. \tag{3.27}$$
$$\left. \frac{1}{C_{i\phi_i^r k}} \sum_{j=1}^{J} \left((R_{\phi_i} X_i^*)_j \overline{W}_{kj} - A_{kj}^{w*} (R_{\phi_i^r} W_i)_j \right)^2 \, d\phi \right)^{\frac{1}{2}}.$$

Modellparameter α. Wir berechnen die partielle Ableitung für den Modellparameter α. Wir setzen (3.20) in (3.24) ein. Für die Optimierung berücksichtigen wir die Randbedingung $\sum_k \int_{\phi^r} \alpha_{k,\phi^r} \, d\phi^r = 1$ aus (Gl. 3.21) und beziehen diese unter Verwendung des *Lagrange*-Multiplikators mit in die zu optimierende Funktion für die untere Grenze (Gl. 3.24) ein. Wir optimieren nun $Z(\Theta; \Theta^{(t)}) + \lambda(1 - \sum_k \int_{\phi^r} \alpha_{k,\phi^r})$:

$$Z(\Theta; \Theta^{(t)}) + \lambda \left(1 - \sum_k \int_{\phi^r} \alpha_{k,\phi^r} \, d\phi^r \right)$$
$$= \sum_{i=1}^{N} \sum_{k=1}^{K} \int_{\phi^r} \int_{\phi^t} P(k, \phi | X_i, \Theta^{(t)}) \Big(log\left(P(X_i | k, \phi, \Theta)\right) + log\left(P(k, \phi^r | \Theta)\right)$$
$$+ log\left(P(\phi^t | k, \phi^r, \Theta)\right) \Big) \, d\phi^t d\phi^r + \lambda \left(1 - \sum_k \int_{\phi^r} \alpha_{k,\phi^r} \, d\phi^r \right).$$

$P(k, \phi^r | \Theta)$ ist in (3.21) definiert und wird entsprechend ersetzt:

$$= \sum_{i=1}^{N} \sum_{k=1}^{K} \int_{\phi^r} \int_{\phi^t} P(k, \phi | X_i, \Theta^{(t)}) \Big(log\left(P(X_i | k, \phi, \Theta)\right) + log\left(\alpha_{k,\phi^r}\right)$$
$$+ log\left(P(\phi^t | k, \phi^r, \Theta)\right) \Big) \, d\phi^t d\phi^r + \lambda \left(1 - \sum_k \int_{\phi^r} \alpha_{k,\phi^r} \, d\phi^r \right).$$

Die Terme $log\left(P(X_i | k, \phi, \Theta)\right)$ und $log\left(P(\phi^t | k, \phi^r, \Theta)\right)$ sind von α unabhängig (Gl. 3.19 und Gl. 3.22). Für die Optimierung berücksichtigen wir die

Randbedingung aus (Gl. 3.21) und bilden die partielle Ableitung nach α_{k,ϕ^r}

$$\frac{\partial Z}{\partial \alpha_{k,\phi^r}} = 0$$

$$= \sum_{i=1}^{N} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)}) \frac{1}{\alpha_{k,\phi^r}}\, d\phi^t - \lambda$$

$$\lambda = \sum_{i=1}^{N} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)}) \frac{1}{\alpha_{k,\phi^r}}\, d\phi^t. \tag{3.28}$$

Wir multiplizieren beide Seiten mit α_{k,ϕ^r}:

$$\lambda\ \alpha_{k,\phi^r} = \sum_{i=1}^{N} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\ d\phi^t.$$

Beide Seiten werden mit $\sum_{k=1}^{K} \int_{\phi^r} .$ integriert:

$$\sum_{k=1}^{K} \int_{\phi^r} \lambda\ \alpha_{k,\phi^r}\ d\phi^r = \sum_{i=1}^{N} \sum_{k=1}^{K} \int_{\phi^r} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\ d\phi^t.$$

Es gilt $\sum_{k=1}^{K} \int_{\phi^r} \alpha_{k,\phi^r}\ d\phi^r = 1$:

$$\lambda = \sum_{i=1}^{N} \sum_{k=1}^{K} \int_{\phi^r} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\ d\phi^t.$$

Wir ersetzen λ durch (Gl. 3.28) und lösen nach α_{k,ϕ^r}:

$$\alpha_{k,\phi^r} = \frac{\sum_{i=1}^{N} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\ d\phi^t}{\sum_{i=1}^{N} \sum_{k=1}^{K} \int_{\phi^r} \int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\ d\phi^r d\phi^t}.$$

Unter Berücksichtigung von $\sum_{i=1}^{N}\sum_{k=1}^{K}\int_{\phi^t}\int_{\phi^t} P(k,\phi|X_i,\Theta^{(t)})\; d\phi^r d\phi^t = N$ erhalten wir den Modellparameter $\alpha_{k,\phi^r}^{(t+1)}$ für den E-Schritt in der folgenden Iteration:

$$\alpha_{k,\phi^r}^{(t+1)} = \frac{1}{N}\sum_{i=1}^{N}\int P(k,\phi^r,\phi^t|X_i,\Theta^{(t)})\; d\phi^t. \tag{3.29}$$

Modellparameter ξ. Wir berechnen die partielle Ableitung für den Modellparameter ξ. Dafür setzen wir (3.22) in (3.24) ein und erhalten:

$$\begin{aligned}
&Z(\Theta;\Theta^{(t)})\\
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\Bigg(logb\left(P(X_i|k,\phi,\Theta)\right) + log\left(P(k,\phi^r|\Theta)\right)\\
&\quad + log\left(P(\phi^t|k,\phi^r,\Theta)\right)\Bigg)\; d\phi.
\end{aligned}$$

$P(\phi^t|k,\phi^r,\Theta)$ entnehmen wir aus (3.22):

$$\begin{aligned}
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\Bigg(log\left(P(X_i|k,\phi,\Theta)\right) + log\left(P(k,\phi^r|\Theta)\right)\\
&\quad + log\left(\frac{1}{(2\pi)^{\frac{3}{2}}\xi^3}\exp\left(-\frac{b_x^2+b_y^2+b_z^2}{2\xi^2}\right)\right)\Bigg)\; d\phi
\end{aligned}$$

$$\begin{aligned}
&= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\Bigg(log\left(P(X_i|k,\phi,\Theta)\right) + log\left(P(k,\phi^r|\Theta)\right)\\
&\quad -\frac{3}{2}\,log(2\pi) - 3\,log(\xi) - \frac{b_x^2+b_y^2+b_z^2}{2\xi^2}\Bigg)\; d\phi.
\end{aligned}$$

Die Terme $log\left(P(X_i|k,\phi,\Theta)\right)$ und $log\left(P(k,\phi^r|\Theta)\right)$ sind von ξ unabhängig (3.19) und (3.21) und die partielle Ableitung nach ξ ergibt sich damit wie folgt:

$$\frac{\partial Z}{\partial \xi} = 0$$
$$= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\left(-\frac{3}{\xi}+\frac{b_x^2+b_y^2+b_z^2}{\xi^3}\right)\,d\phi.$$

Wir multiplizieren beide Seiten mit ξ^3 :

$$= \sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\left(-3\;\xi^2+b_x^2+b_y^2+b_z^2\right)\,d\phi.$$

$-\,3\;\xi^3$ ist ein konstanter Faktor und kann vor das Integral und die Summen gezogen werden. Durch ausmultiplizieren der Klammer erhalten wir:

$$= -\;3\;\xi^2\sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\,d\phi$$
$$+\sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\;\left(b_x^2+b_y^2+b_z^2\right)d\phi.$$

Wir verwenden

$$\sum_{i=1}^{N}\sum_{k=1}^{K}\int P(k,\phi|X_i,\Theta^{(t)})\,d\phi = N$$

und lösen die Gleichung nach ξ auf:

$$\xi^{(t+1)} = \left(\frac{1}{3N}\sum_{i=1}^{N}\int P(k,\phi|X_i,\Theta^{(t)})\,(b_x^2+b_y^2+b_z^2)\,d\phi\right)^{\frac{1}{2}}. \tag{3.30}$$

Durch die gezeigten Ableitungen sind die Berechnungsvorschriften für die vier Modellparameter $\Theta = (A_k^w,\ \alpha_{k,\phi^r},\ \sigma,\ \xi)$ für jede Iteration vollständig gegeben.

Sind die Berechnungen im M-Schritt abgeschlossen, beginnt der Algorithmus mit dem E-Schritt der nächsten Iteration und verwendet die neu berechneten Modellparameter. Die ML-Optimierung iteriert nun schrittweise bis sie konvergiert, d.h. bis sich der *Likelihood* nicht mehr verbessert und sich die Modellparameter von einer Iteration zur nächsten nicht mehr verändern. Der *Compound Wedge* $\overline{W}_k$ bleibt dabei konstant. Allerdings hängt die Zusammensetzung des *Compound Wedge* von der Orientierung der Partikel ab und ist zu Beginn der Klassifikation nicht bekannt. Deswegen durchläuft der Algorithmus typischerweise eine äußere Schleife (Abb.: 3.7), um eine Aktualisierung durchzuführen. Normalerweise wird die Optimierung mit einem gleichverteilten *Compound Wedge* gestartet und die entsprechende Log-*Likelihood*-Funktion in der inneren Schleife optimiert. Nachdem das ML-Verfahren konvergiert ist oder nach einer definierten Anzahl an Iterationen wird der *Compound Wedge* $\overline{W}_k^{(\text{new})}$ wie folgt neu berechnet:

$$\overline{W}_k^{(\text{new})} = \frac{\sum_{i=1}^{N} \int P(k, \phi | X_i, \Theta^{(t)}) \; (R_{\phi^r} W_i) \; d\phi}{\sum_{i=1}^{N} \int P(k, \phi | X_i, \Theta^{(t)}) \; d\phi} \; . \tag{3.31}$$

Liegt ein Unterschied zwischen dem neuen und dem alten $\overline{W}_k$ vor, so wird die ML-Optimierung mit $\overline{W}_k^{(\text{new})}$ neu gestartet. Diese äußere Schleife wird wiederholt, bis $\overline{W}_k$ sich nicht mehr verändert. Häufig ist dies nach drei Iterationen durch die äußere Schleife der Fall (Abb.: 3.7).

3.7 Abtastung der Rotationen im dreidimensionalen Raum

Die Positionen und Orientierungen von makromolekularen Strukturen sind in den aufgezeichneten Tomogrammen unbekannt. Die richtige Translation und Rotation der Partikel in den Subtomogrammen zu identifizieren, ist ein grundlegendes Ziel der in diesem Kapitel vorgestellten Analyseverfahren. Für die Positionsbestimmung werden alle möglichen Translationen voxelgenau abgetastet. Die Abtastung der dreidimensionalen Rotationen hingegen ist ein skalierbarer Parameter, der explizit für jede Analyse definiert werden muss. Je feiner die Abtastung, umso besser ist die Auflösung der Orientierung der Partikel. Bei der Analyse von Subtomogrammen mit der ML-Klassifikation gehen alle Partikelorientierungen entsprechend ihrer Wahrscheinlichkeit in das Klassenmitttel ein (Abschn.: 3.6). Ist die Abtastung nicht gleich verteilt, kommt es zu einer unerwünschten Verstärkung feiner abgetasteter Bereiche in der Mittelung. Hinzu kommt, dass die Anzahl der abgetasteten Rotationen als Faktor in die Laufzeit und ggfs. in den Speicherbedarf eingeht. Aus

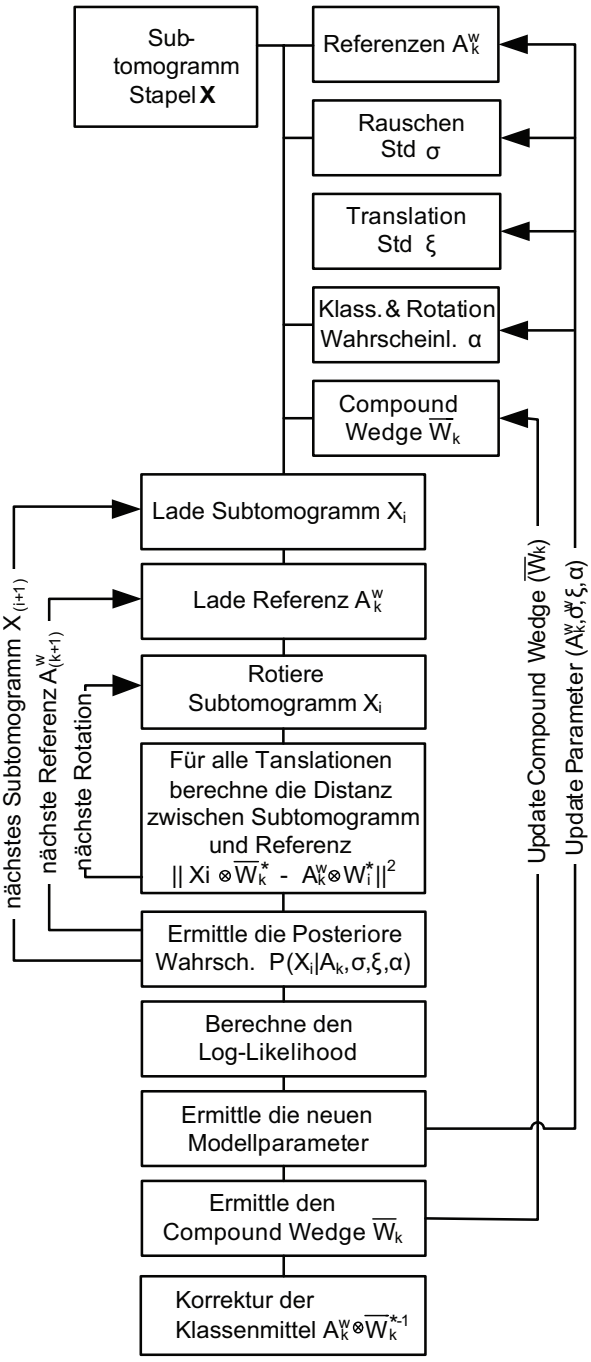


Abbildung 3.7: Flussdiagramm von MLTOMO

diesem Grund ist es sinnvoll und notwendig, die Rotationen möglichst gleichmäßig zu verteilen, um die Wahrscheinlichkeiten richtig zu berechnen und ein optimales Verhältnis von Abtastung zur Anzahl der Rotationen zu erzeugen.

3.7.1 Mathematische Beschreibung der Rotationen

Die Rotation ist eine Drehbewegung eines Objektes. Im Gegensatz zur reinen Translation ist die Rotation keine Bewegung, die den Schwerpunkt des Körpers durch den Raum bewegt, sondern eine Drehung des Körpers um eine Rotationsachse. Alle Punkte, die genau auf dieser Achse liegen, verändern ihre Position nicht. Alle anderen Punkte bewegen sich auf einer idealen Kreisbahn mit einem Radius, der dem Abstand zur Rotationsachse entspricht. Im dreidimensionalen Raum ist eine Rotation deswegen mit einem Winkel und einer Rotationsachse vollständig beschrieben. Mathematisch kann eine Rotation als orthogonale Transformation zwischen zwei Vektorräumen verstanden werden. Diese lineare Transformation kann im Euklidischen Raum durch Rotationsmatrizen eindeutig beschrieben werden.

Rotationsmatrizen. Rotationsmatrizen sind reelle quadratische Matrizen R. Im dreidimensionalen Euklidischen Raum kann ein Punkt p durch die Multiplikation mit der Matrix rotiert werden: $p' = R \cdot p$. R ist eine orthogonale Matrix, die durch $det(R) = 1$ beschränkt ist und für die das inverse Element gleich der Transponierten: $RR^T = R^T R = I$ definiert ist. Die Einheitsmatrix I entspricht dabei einer Null-Rotation. Nacheinander ausgeführte Rotationen können durch Multipikation ihrer Rotationsmatrizen verkettet werden: $R_2 \cdot R_1 p = (R_2 R_1) \cdot p$. Eine Rotation ist durch R eindeutig beschrieben.

Eulerwinkel. Eulerwinkel sind eine häufig verwendete Möglichkeit zur Beschreibung der Orientierung von Objekten im dreidimensionalen Raum. Es handelt sich um drei Winkel, welche jeweils eine Rotation um bestimmte Achsen beschreiben, und so eine Transformation zwischen zwei kartesischen Koordinatensystemen definieren. Es existieren verschiedene Definitionen für die Eulerwinkel, die sich in der Wahl der Drehachsen unterscheiden. Die hier verwendete Konvention ist Z,X',Z'' (Abb.: 3.8). Zuerst wird mit einem Winkel φ um die z-Achse des globalen Koordinatensystems (Z) gedreht. Es folgt eine Rotation mit dem Winkel θ um die neue x-Achse (X') und schließlich mit dem Winkel ψ um die nach den beiden vorherigen Drehungen erhaltene z-Achse (Z''). Die Rotationen um eine Koordinatenachse kön-

nen durch folgende Rotationsmatrizen dargestellt und hintereinander ausgeführt werden:

$$
\begin{aligned}
R_{(\varphi,\theta,\psi)} &= R_{\mathsf{Z},\psi} \cdot R_{\mathsf{X},\theta} \cdot R_{\mathsf{Z},\phi} \\
R_{\mathsf{Z},\psi} &= \begin{pmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \\
R_{\mathsf{X},\theta} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos(\theta) & sin(\theta) \\ 0 & -sin(\theta) & cos(\theta) \end{pmatrix} \\
R_{\mathsf{Z},\phi} &= \begin{pmatrix} cos(\varphi) & sin(\varphi) & 0 \\ -sin(\varphi) & cos(\varphi) & 0 \\ 0 & 0 & 1 \end{pmatrix}.
\end{aligned}
$$

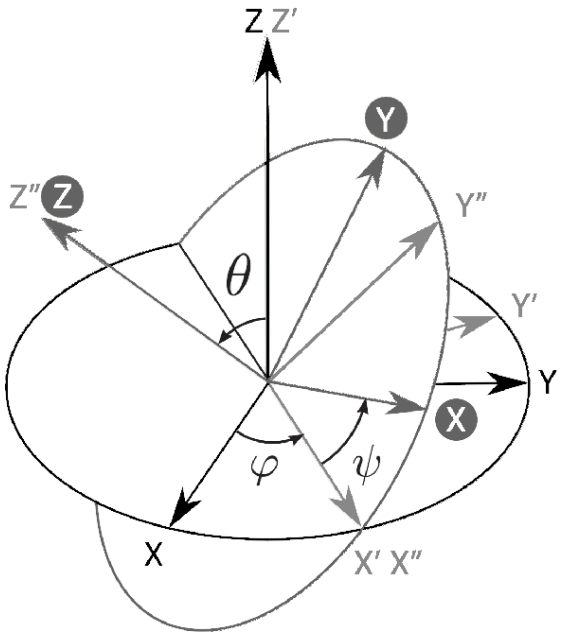


Abbildung 3.8: Beschreibung der Rotation durch Eulerwinkel (φ, θ, ψ) um die Achsen Z, X', Z''. In Schwarz das Ausgangs-Koordinatensystem und in dunkelgrau mit weißen Buchstaben das resultierende Koordinatensystem.

Alle möglichen Orientierungen im Raum lassen sich so mit $\varphi \in \{0..360°\}$, $\theta \in \{0..180°\}$ und $\psi \in \{0..360°\}$ beschreiben. Im Gegensatz zu den Rotationsmatrizen ist die Abbildung von Eulerwinkeln auf Rotationen surjektiv.

Einheits-Quaternionen. Quaternionen sind eine algebraische Struktur, mit deren Hilfe sich Rotationen beschreiben lassen [Vicci, 2001] [Kuipers, 2002]. Sie sind eine Erweiterung der komplexen Zahlen und wurden 1843 von Sir William Rowan Hamilton [Hamilton, 1844] erdacht. Quaternionen sind als nicht kommutativer Ring definiert.

$$Q = \{a + bi + cj + dk | a, b, c, d \in \mathbb{R}\} \tag{3.32}$$

Der Additionsoperator ist definiert als:

$$(a_1+b_1i+c_1j+d_1k)+(a_2+b_2i+c_2j+d_2k) = (a_1+a_2)+(b_1+b_2)i+(c_1+c_2)j+(d_1+d_2)k \tag{3.33}$$

Die Multiplikation kann durch folgenden Term auf Basis des Ditributionsgesetzes erfolgen:

$$(a_1 + b_1i + c_1j + d_1k)(a_2 + b_2i + c_2j + d_2k) \tag{3.34}$$

Die Symbole i, j und $k \in \mathbb{R}$ sind abstrakte Symbole und ähnlich der imaginären Einheiten der komplexen Zahlen $\mathbb{C}$. Die Multiplikation ist nicht kommutativ und folgende Regeln gelten:

$$i^2 = j^2 = k^2 = ijk = -1$$

ij = k,	ji = -k
jk = i,	kj = -i
ki = j,	ik = -j.

Jedes Quaternion ist eine lineare Kombination der orthogonalen Basisvektoren 1, i, j und k. Wie bei den komplexen Zahlen spricht man von einem Real- und Imaginärteil. Die Konjugation ist definiert als $q^* = a - bi - cj - dk$. Der Betrag ist die Euklidische Norm der vier Elemente $|q| = \sqrt{a^2 + b^2 + c^2 + d^2} = \sqrt{qq^*}$. Es ist üblich, den Imaginärteil als Vektor zu betrachten und den Realteil als Skalar. Dadurch lässt sich die Multiplikation der Quaternionen als Vektor- und Kreuzprodukt verstehen:

$$(a + \mathtt{v})(b + \mathtt{w}) = (ab - \mathtt{vw}) + (a\mathtt{w} + b\mathtt{v} + \mathtt{v} \times \mathtt{w}). \tag{3.35}$$

Dies führt zum multiplikativen Inversen $q = (a + \mathtt{v})$:

$$q^{-1} = (a + \mathtt{v})^{-1} = \frac{a - \mathtt{v}}{s^2 + |\mathtt{v}|^2} = \frac{q^*}{|q|^2}.$$

3.7.2 Naives Abtastungsschema

Intuitiv lässt sich ein Abtastungsschema für die Orientierungen von Partikeln erzeugen, indem die drei Eulerwinkel nacheinander um einen festen Wert inkre-

mentiert werden. Das dabei optimierte Distanzmaß korrespondiert mit der Euklidischen Distanz zwischen zwei Tripeln von Eulerwinkeln $\phi_i^r = (\varphi_i, \theta_i, \psi_i)$ mit $i = 1, 2$

$$d(\phi_1^r, \phi_2^r) = \sqrt{\Delta(\varphi_1, \varphi_2)^2 + \Delta(\theta_1, \theta_2)^2 + \Delta(\psi_1, \psi_2)^2}. \tag{3.36}$$

Dabei ist $\Delta(\alpha_1, \alpha_2)$ die kleinste Distanz zwischen zwei Winkeln bezüglich ihrer Periodizität von 2π.

Ein Inkrement von z.B. 90° erzeugt die in Tabelle 3.7.2 gezeigten 32 Eulerwinkel. Wie bereits erwähnt, können zwei unterschiedliche Eulerwinkel dieselbe Rotation beschreiben. In unserem Beispiel sind 12 von 32 Winkeln redundant. Dies lässt sich dadurch beheben, dass $\theta \neq 0°$ und $\theta \neq 180°$ gewählt wird. Trotzdem werden spätere Analysen zeigen, dass sich bei dieser Methode eine ungleiche Verteilung der Winkel ergibt.

3.7.3 Algorithmus zur Näherung einer äquidistanten Abtastung

Rotationen lassen sich durch Einheits-Quaternionen mit dem Betrag eins darstellen. Betrachtet man die Quaternionen als vierdimensionalen Vektor, sind es alle Vektoren, die die Oberfläche einer vierdimensionalen Einheits-Hyperkugel beschreiben. Aus diesem Grund lassen sich die Einheitsquaternionen definieren als $q = cos(\alpha/2) + sin(\alpha/2)\mathtt{v}$. α ist der Winkel der Rotation und der Vektor $\mathtt{v}$ ist die Richtung der Rotationsachse, normalisiert auf die Länge eins. Es kann gezeigt werden, dass die Rotation eines dreidimensionalen Punktes $\mathtt{p}$ mit einem Winkel α um die Achse $\mathtt{v}$ berechnet werden kann als $\mathtt{p}' = q\mathtt{p}q^{-1} (= q\mathtt{p}q^*)$. Die nacheinander Ausführung von Rotationen q_1 und q_2 wird durch das Produkt $q_2 q_1$ beschrieben, weil $(q_2 q_1)\mathtt{p}(q_2 q_1)^{-1} = q_2(q_1 \mathtt{p} q_1^{-1})q_2^{-1}$ gilt. Daraus ergibt sich die inverse Rotation von q als q^{-1}. Gegenüberliegende Punkte auf der Hyperkugel beschreiben die gleiche Rotation $qpq^-1 = (-q)p(-q)^-1$.

Die Quaternionen erlauben es, ein einfaches Abstandsmaß zwischen Rotationen zu definieren. Diese Eigenschaft wird vor allem in der Computergrafik benutzt, wenn zwischen zwei Rotationen interpoliert werden soll. Der kürzeste Weg entlang der Oberfläche der Hyperkugel ergibt die optimale Bewegung von einer Orientierung in die andere mit konstanter Winkelgeschwindigkeit [Dam et al., 1998].

Das Ziel des Algorithmus ist es, den Rotationenraum möglichst gleichmäßig zu verteilen, d.h. dass der Abstand über alle Rotationen zu den benachbarten Rota-

Tabelle 3.1: Gleichverteilung von Eulerwinkeln mit einem Inkrement von 90°. In der vierten Spate sind redundante Winkelsätze markiert.

Tripel	φ	θ	ψ	gleich zu Tripel
1	0	0	0	12
2	90	0	0	9
3	180	0	0	10
4	270	0	0	11
5	0	90	0	
6	90	90	0	
7	180	90	0	
8	270	90	0	
9	0	0	90	2
10	90	0	90	3
11	180	0	90	4
12	270	0	90	1
13	0	90	90	
14	90	90	90	
15	180	90	90	
16	270	90	90	
17	0	0	180	3
18	90	0	180	4
19	180	0	180	1
20	270	0	180	2
21	0	90	180	
22	90	90	180	
23	180	90	180	
24	270	90	180	
25	0	0	270	4
26	90	0	270	1
27	180	0	270	2
28	270	0	270	3
29	0	90	270	
30	90	90	270	
31	180	90	270	
32	270	90	270	

tionen möglichst gleich sein soll. Für jeden Algorithmus wurde deswegen zuerst ein Distanzmaß definiert und danach eine Strategie, dieses Distanzmaß im Sinne der Gleichverteilung zu optimieren.

Dreidimensionale Rotationen, dargestellt durch Einheits-Quaternionen, können verwendet werden, um zwischen Rotationen entlang der Oberfläche einer vierdimensionalen Einheitskugel zu interpolieren. Wie im dreidimensionalen Fall kann als Abstandsmaß das Skalarprodukt der Quaternionen-Koeffizienten von zwei Rotationen gewählt werden, das über den Winkel proportional ist zum geodätischen Pfad auf der Einheits-Hyperkugel. Allerdings muss berücksichtigt werden, dass zwei gegenüberliegende Quaternionen auf der Hyperkugel identische Rotationen darstellen:

$$q_2 \cdot q_1 = |q_2||q_1|cos(\alpha). \tag{3.37}$$

Um auf Basis der Quaternionen eine äquidistante Verteilung der Rotationen zu erhalten, wurden in der Arbeit von Plaisier [Plaisier et al., 2007] die Ecken vierdimensionaler platonischer Körper berechnet von denen aus die Abtastung verfeinert wurde [Karney, 2007]. In dieser Arbeit wurde für die Verteilung der Rotationen ein Kraftfeld entwickelt. Um n Rotationen zu verteilen, wurden n Punkte auf einer vierdimensionalen Hyperkugel verteilt. Jeder Punkt entspricht einem Quaternion. Die Kraft F_i in jedem Punkt p_i berechnet sich auf Basis des Euklidischen Abstands zu allen anderen Punkten p_j mit $i, j \in \{1, 2, ..., n\}$

$$-F_i = \sum_{j \neq i} \left(\frac{p_j - p_i}{|p_j - p_i|} \cdot w(|p_j - p_i|) + \frac{-p_j - p_i}{|-p_j - p_i|} \cdot w(|-p_j - p_i|) \right). \tag{3.38}$$

Die Wichtungsfunktion w ist invers proportional zum Abstand zwischen p_i und p_j

$$w(|p_j - p_i|) = \frac{1}{|p_j - p_i|^2}. \tag{3.39}$$

Die Kraft, die auf den Punkt p_i wirkt, ergibt sich aus allen anderen Punkten p_j, aber auch gleichzeitig aus deren antipodischen Punkten $-p_j$. Der Kraftvektor zeigt von der Oberfläche der Hyperkugel fort, daher muss der Punkt, nachdem er in diese Richtung bewegt wurde, wieder auf die Oberfläche zurückprojiziert werden und erhält so seine neue Position. Zeigen alle Kraftvektoren in allen Punkten in Richtung der Oberflächennormalen, ist ein globales Optimum gefunden und das System ist im Kräftegleichgewicht. Um eine zu große Schrittweite zu vermeiden, wird zu Beginn der Optimierung die Kraft mit einem Skalierungsfaktor multipliziert, der mit fortschreitender Anzahl an Iterationen verringert wird. Die resultierende Liste wird

dann in Eulerwinkel umgerechnet, um für die weiteren Verfahren zu Verfügung zu stehen.

3.7.4 Bewertung des Verteilungsalgorithmus

Ein Winkelinkrement von 15° ist eine übliche Abtastung bei der Analyse von Tomogrammen. Bei einer Gleichverteilung der Eulerwinkel wie im gezeigten Bespiel 3.7.2 sind dies 7488 Eulerwinkel. Bei der Verteilung von Quaternionen auf einer Einheits-Hyperkugel unter Verwendung des Kraftfeldes werden 7112 Rotationen benötigt, um ein maximales Winkelinkrement von 15° zu erreichen.

Zwei unterschiedliche Messverfahren bewerteten die Ergebnisse der Berechnungen. Für eine optimale Verteilung sollten benachbarte Rotationen möglichst denselben Winkelabstand besitzen. Deswegen ermittelt das erste Messverfahren von jeder Rotation den Winkelabstand zur nächsten benachbarten Rotation.

Theoretisch ist es möglich, dass alle Rotationen den gleichen Abstand zum nächsten Nachbarn besitzen und trotzdem ganze Bereiche des Rotationsraumes leer sind. Um diese Möglichkeit auszuschließen, berechnet das zweite Messverfahren die Volumen der Delaunay-Tetraeder auf der Oberfläche der vierdimensionalen Einheits-Hyperkugel [Eppstein, 1992]. Dabei werden die Eulerwinkel in Quaternionen überführt und die Delaunay-Triangulierung der Punkte auf der Hyperkugel berechnet. Die Triangulierung erzeugt vierdimensionale Volumen, bestehend aus fünf Eckpunkten und vier Seiten. Um zu erkennen, wie dicht der Raum der Rotationen abgetastet wurde, wird das Volumen jener dreidimensionalen Seiten berechnet, welche direkt unter der Oberfläche der Hyperkugel liegen. Im optimalen Fall sind alle Oberflächenvolumen gleich groß.

Die Messergebnisse für die lineare Verteilung der Eulerwinkel (Abb.: 3.9) weisen eine ungleiche Verteilung der Rotationen auf. Der minimale Winkelabstand reicht von 3° bis 15°. Die Oberflächenvolumen zeigen ein ähnliches Bild.

Bei den Histogrammem zu den Ergebnissen der Kraftfeldoptimierung (Abb.: 3.10 c und d) reicht der minimale Winkelabstand von 12.6° bis 13.0° und auch die Oberflächenvolumen deuten auf keine feiner oder gröber abgetasteten Bereiche hin.

Die Kraftfeldoptimierung ist laufzeitintensiv. Für n Winkel ergibt sich eine Laufzeit $O(n^2)$, da der Abstand von jedem Winkel zu jedem anderen berechnet werden

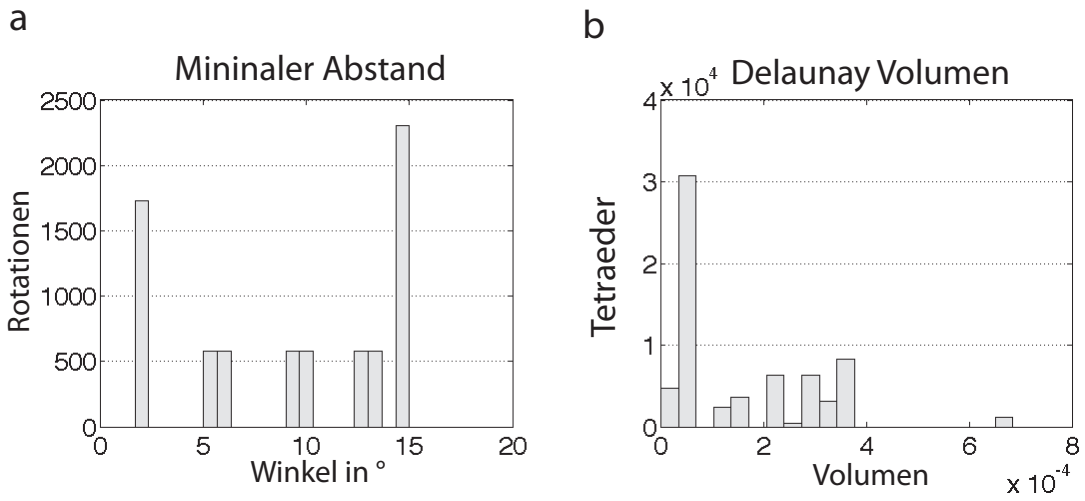


Abbildung 3.9: Messung zur Verteilung von 7488 Eulerwinkeln. Eulertripel, die dieselbe Rotation beschreiben, wurden bei der Berechnung vermieden ($\theta \neq 0°$,$\theta \neq 180°$). a) Histogramm über den Winkelabstand zur nächsten benachbarten Rotation b) Histogramm über das Volumen der Delaunay-Tetraeder auf der Oberfläche der vierdimensionalen Einheits-Hyperkugel [Stölken, 2008].

muss. Der Algorithmus rechnet ca. einen Tag auf einem Standardprozessor, bis das Ergebnis die hier gezeigte Qualität erreicht.

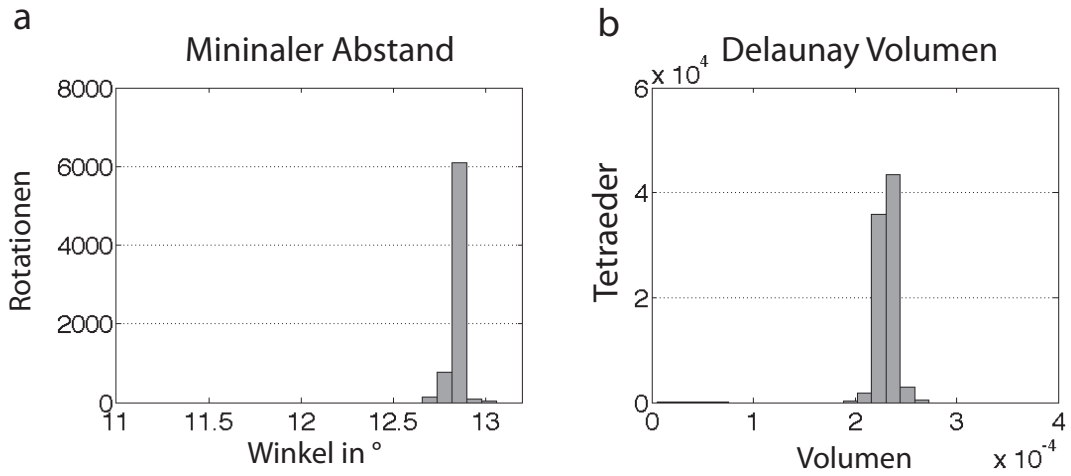


Abbildung 3.10: Messung zur Verteilung von Quaternionen auf einer Einheits-Hyperkugel unter Verwendung eines Kraftfeldes a) Histogramm über den Winkelabstand zur nächsten benachbarten Rotation. b)Histogramm über das Volumen der Delaunay Tetraeder auf der Oberfläche der vierdimensionalen Einheits-Hyperkugel [Stölken, 2008].

4 Implementierung

Bei den vorgestellten Algorithmen zur Analyse von Subtomogrammen (Kapitel 3) werden 3D-Subvolumen aus elektronenmikroskopischen Tomogrammen verarbeitet. Die Volumen bestehen aus Voxeln, die jeweils einen Grauwert im Bild beschreiben. Je nach Auflösung des Tomogramms und der Größe der zu erwartenden biologischen Komplexe, können Sutomogramme eine Volumengröße bis zu 512x512x512 Voxel haben. In Kombination mit den durchgeführten Operationen auf den Volumen ergibt sich ein erheblicher Berechnungsaufwand für die Anwendung der Analyseverfahren. In diesem Kapitel wird auf die konkrete Implementierung der Algorithmen eingegangen. Um die Berechenbarkeit in annehmbarer Zeit zu gewährleisten, ist es notwendig, die Algorithmen zu parallelisieren. Insbesondere bei der Implementierung des ML-Algorithmus sind Randbedingungen zu beachten, die einen naiven Ansatz zur parallelen Verarbeitung nicht zulassen. Namen für Variablen, Funktionen und Zustände, die in diesem Kapitel verwendet werden, entsprechen häufig der Bezeichnung im Quelltext des C++ Programmes. Die Verwendung der gleichen Nomenklatur soll das Lesen der Quellen erleichtern.

4.1 Architektur und Programmbibliotheken

Als Implementierungssprache wurde C++ gewählt. Die Programmiersprache ist eine geeignete Wahl, da sie einerseits auf den anvisierten Zielplattformen kompiliert werden kann und eine möglichst effiziente, native Implementierung ermöglicht. Als Compiler wurde dabei der GNU C++-Compiler auf einem GNU/Linux-*Computer-Cluster* (Max Planck Institut für Biochemie, Martinried, Deutschland) und der IBM CL C/C++-Compiler auf einem IBM *Blue-Gene* System (AIX-Unix) (Rechenzentrum der Max Planck Gesellschaft, Garching, Deutschland) verwendet.

Für das verteilte Rechnen auf Basis mehrerer Prozesse wurde das *Message Passing Interface* (MPI) verwendet. Weitere verwendete Programmbibliotheken sind die FFTW3-Bibliothek [Frigo and Johnson, 2005] zur Berechnung der schnellen

diskreten Fouriertransformation und Teile der Boost-Bibliothek, welche allgemeine C++ Hilfsklassen bereitstellt.

Darüber hinaus wurde auf eine im Institut entwickelte C++-Bibliothek (*libtomc*) eingebunden [Haller, 2008]. Diese Bibliothek beinhaltet allgemeine Hilfsroutinen für tomographische Daten, wie etwa zur Transformation von Partikel-Volumen und dem Lesen und Schreiben des EM-Dateiformats [Hegerl, 1996].

Alle verwendeten Bibliotheken sind unter einer freien Software-Lizenz erhältlich. Die parallelisierte Implementierung des ML Algorithmus (MLTOMO) kann von der Webseite `http://www.biochem.mpg.de/mltomo` heruntergeladen oder vom Autor angefordert werden.

4.1.1 MPI - *Message Passing Interface*

Erst durch eine parallele Berechnung der Ergebnisse ist es überhaupt möglich, einen rechenintensiven Algorithmus in akzeptabler Zeit auf reale Datensätze anzuwenden. Für die Verteilung der Berechnungen auf mehrere Prozesse wurde das MPI verwendet. MPI ist eine Applikations-Schnittstellen-Spezifikation und ein Defakto-Standard für paralleles Rechnen auf verteilten Systemen.

Dabei handelt es sich erstmal um eine Schnittstelle, für welche eine Reihe verschiedener Implementierungen existiert. Konkret wurde in der Programmumgebung mit den C-Bindings der OpenMPI-Bibliothek entwickelt. Damit ist die Portierung auf eine andere, standard-konforme MPI-Implementierung problemlos möglich. MPI ermöglicht die parallele Ausführung des Algorithmus auf meheren Computern oder einem *Computer-Cluster*. Von Anfang an wurde auch darauf abgezielt, das Programm auf dem *Blue-Gene* System des Rechenzentrums der Max Planck Gesellschaft in Garching rechnen zu lassen. Dabei handelt es sich um einen *Computer-Cluster* mit mehreren 1000 Prozessoren.

Bei MPI-Applikationen startet die MPI-Umgebung eine Menge von Prozessen auf meist separaten Rechnern oder Rechner-Knoten. Dabei führen die Prozesse im Normalfall dasselbe Programm aus und verwenden ein gemeinsames Dateisystem von dem Daten gelesen und geschrieben werden. Wesentlich dabei ist, dass die einzelnen Prozesse über keinen gemeinsamen Speicher verfügen, sondern die Kommunikation durch den Austausch von Nachrichten stattfindet (*Message passing*). MPI definiert Funktionen zur Synchronisation und dem Austausch von strukturierten Daten unter den Prozessen. Weiterhin werden die verwendeten MPI-Prozesse

nicht in weitere Teil-Prozesse (*Threads*) aufgesplittet. Die Parallelisierung findet über Prozesse statt und nicht durch *Threads*, denn auf Systemen wie dem *BlueGene* können *Threads* nicht nach Belieben gestartet werden.

Nachdem alle MPI-Prozesse starten, rufen sie eine Initialisierungs-Routine auf (`MPI_Init`). Dabei erhält jeder Prozess unter anderem eine eindeutige Kennnummer (*Rank*) und die Anzahl der anderen Prozesse in der derzeitigen Rechenumgebung. Anhand dieses *Rank*s kennen die einzelnen Prozesse ihre teils unterschiedlichen Aufgaben und nehmen sie wahr.

4.1.2 Aufbau der Implementierung

Der Aufbau der Implementierung besteht aus zwei wesentlichen Bausteinen: der Basisbibliothek *libtomc* und des Klassifikations- und Alignierungsprogramms, das die Funktionen der Bibliothek verwendet. *libtomc* stellt die Kernfunktionen für die Verarbeitung der tomographischen Daten bereit. Sie besteht aus C++ Klassen, die grundlegende Funktionalitäten für Volumen bzw. 3D-Arrays bieten:

- Effiziente Funktionen zum Lesen und Scheiben von Tomogrammen und Subtomogrammen im EM-Format
- Kopieren von Volumen
- Ausschneiden von Subvolumen
- Elementweise Operatoren auf Volumen mit einem Skalar (Addition, Multiplikation etc.)
- Elementweise Operatoren auf Volumen mit einem anderen Volumen (Addition, Multiplikation etc.)
- Schnelle Rotationen mit unterschiedlichen Interpolationen zwischen den Voxel (*Naerest Neighbor*, Trilinenare Interpolation, Spline Interpolation)
- Berechnung der schnellen Fouriertransformation mit der FFTW-3
- Erstellen von Masken
- Reduzierieren der Volumengröße (*Binning*)

- Unterschiedliche Filtermöglichkeiten
- Verschiedene Darstellungen des *Missing Wedge* (speichereffizienter *Single Axis Wedge*, *Wedge* Volumen)

Zusätzlich erlaubt die Basisbibliothek das Versenden von Volumen per MPI und stellt damit die Grundlage für die Interprozesskommunikation für paralleles Rechnen dar.

libtomc wurde gemeinsam mit einer Implementierung für die Multi-Referenz-Klassifikation (Abschn.: 3.5) von Haller entwickelt und auf Effizienz und Geschwindigkeit optimiert [Haller, 2008]. TOMCORR3D war die erste Implementierung auf Basis der *libtomc* Bibliothek. Die im Folgenden beschriebene Implementierung von MLTOMO setzt ebenfalls auf *libtomc* auf und nutzt dessen Basisfunktionalität (Abb.: 4.1).

4.2 Der parallelisierte ML-Algorithmus

Um die Berechnung effizient zu gestalten, wurde eine Strategie für die Parallelisierung des ML-Algorithmus entwickelt, die es erlaubt den E-Schritt und den M-Schritt des EM-Algorithmus in einem Schritt zu berechnen und somit redundante Berechnungen zu vermeiden. Die Skalierbarkeit über eine Vielzahl von Prozessoren wird anhand eines einfachen Beispiels erläutert und bewertet.

4.2.1 Ein- und Ausgabeparameter

Die wichtigsten Eingabeparameter sind der Partikelstapel (Subtomogramme mit Partikelbildern), die initialen Referenzvolumen, die Liste der Rotationen, die initialen Modellparameter und ein Abbruchkriterium. Auf Basis dieser Eingaben berechnet der Algorithmus den *Likelihood* für das angenommene Modell, für jede Klasse eine Mittelung von Subtomogrammen des Partikelstapel und die dazugehörigen Modell- und Alignierungsparameter. Einige wichtige Aspekte der Parameter bzgl. der Implementierung werden hier kurz dargestellt (Detaillierte Informationen über alle Konfigurationsoptionen sind der Programmdokumentation zu entnehmen):

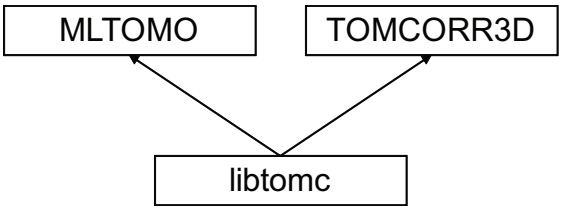


Abbildung 4.1: Sofware Architektur. Die C++ Bibliothek libtomc stellt die Basisfunktionalität auf EM-Volumen bereit. Die Klassifikations- und Alignierungsalgorithmen TOMCORR3D und MLTOMO verwenden die Bibliothek und deren Funktionen, um Subtomogramme berarbeiten zu können.

Eingabeparameter

- **Partikel**. Initial wird eine Liste von Partikeln mit dem jeweiligen *Missing Wedge* im EM-Format benötigt. Bei Programmstart werden die Dateinamen der originalen Partikel als Konfigurationsparameter übergeben. In einem Vorprozessierungsschritt werden die Partikel (optional) normiert, im Frequenzraum gefiltert, und in einem temporären Verzeichnis abgespeichert. Auch eine Kopie aller Partikel-*Wedge* wird erstellt, bei dem die Koeffizenten der einzelnen Frequenzen durch einen FFT-Shift für die Berechnung geeignet angeordnet werden. Im Verlauf der nachfolgenden Iterationen werden immer diese vormodifizierten Partikel verwendet und können von den *Worker*-Prozessen (Abschn.: 4.2.3) bei Bedarf vom Dateisystem geladen werden.

- **Referenzen**. Der Algorithmus benötigt eine Anzahl an unterschiedlichen Start-Referenzen. Die Liste von Referenzen definiert gleichzeitig die Anzahl der Klassen, die über alle Iterationen einer Klassifikation konstant ist. Diese Start-Referenzen werden unmodifiziert als Referenz der ersten Iteration verwendet. Unter verschiedenen Rotationen werden die (vorprozessierten) Partikel mit den Referenzen verglichen. Zusätzlich kann ein *Compound Wedge* angegeben werden. Die Parameter sind insbesondere dann nützlich, wenn nach einem Programmabbruch mit der Klassifikation nach der zuletzt berechneten Iteration fortgefahren werden soll.

- **Rotationen**. Eine Implementierung des ML-Algorithmus kann natürlich nur eine diskrete, endliche Anzahl von Rotationen berechnen. Dazu wird dem Programm eine Liste von möglichst äquidistant-verteilten Rotationen (Orientierungen) mitgegeben. Die Liste wird benötigt, um in jeder Iteration die Partikel verschiedener Orientierungen mit der Referenz zu verglichen. (In

MLTOMO selbst wird die Referenz in die inverse Orientierung gedreht und mit dem unrotierten Partikel verglichen, da dies Vorteile bei er Implementierung hat).

- **Modellparameter**. Die Modellparameter, wie die Varianz des Rauschens (σ) oder die Varianz der Positionierung der Partikel (ξ), gehen als initialer Wert in die Berechnung ein. Je höher die Startwerte der Varianzen eingestellt werden, desto höher wird die Anzahl an möglichen Lösungen mit einer ähnlichen Wahrscheinlichkeit. Der Algorithmus braucht dann länger, um die optimale Lösung zu finden, betrachtet aber auch eine größere Anzahl an Möglichkeiten. Generell ist es sinnvoll, die Startwerte der Varianzen mit großen Werten zu belegen, wenn wenig über die Strukturen in den untersuchten Subtomogrammen bekannt ist bzw. wenn keine initalen Referenzen von bekannten Strukturen oder homologen Strukturen für den Programmstart vorliegen.

- **Abbruchkriterium**. Es kann die Anzahl der Iterationen vorgegeben werden, nach denen das Programm beendet werden soll. Zusätzlich gibt es einen Abbruchparameter in %, der auf der Ähnlichkeit der Mittelung aus der letzten Interation mit der aktuellen Mittelung innerhalb einer Klasse basiert. MLTOMO wird beendet, wenn entweder die Anzahl der eingestellten Iterationen erreicht ist oder die Ähnlichkeit in allen Klassen den eingestellten Schwellenwert erreicht hat.

Ausgabeparameter

- **Likelihood**. Die optimierte Gesamtwahrscheinlichkeit (*Likelihood*) für die gegebenen Eingabeparameter wird in jeder Iteration berechnet und gespeichert. Sie ist ein numerischer Wert, der es zulässt, das Ergebnis unterschiedlicher Läufe auf dem gleichen Partikelstapel quantitativ zu vergleichen.

- **Klassenmittelung**. Für jede Klasse wird nach jeder Iteration das gewichtete Mittel über alle Partikel, Rotationen und Translationen abgespeichert (unkorrigiert). Zusätzlich wird für jede Klasse das korrigierte Mittel (Abschn.: 3.2) gespeichert.

- **Modellparameter**. Alle berechneten Modellparameter werden gespeichert. Da die Parameter die Eingangsparameter (Abb.: 3.7) für die nächste Iteration sind, kann ein Benutzer das Programm vorzeitig abbrechen und jederzeit auf Basis der 'Zwischenergebnisse' von diesem Punkt aus fortsetzen.

- **Alignierungsparameter** Es werden die Alignierungsparameter mit der höchsten Wahrscheinlichkeit und die Klassen-Zugehörigkeitswahrscheinlichkeit zu jedem Referenz-Partikel-Paar (nicht zu jedem Klassenmittel) abgespeichert. Eine Mittelung auf Basis der Alignierungsparameter ist nicht identisch mit der Klassenmittelung, da letztere ein gewichtetes Mittel über alle Rotationen, Translationen und Partikel darstellt. Die Klassenmittel sind die valide Referenz für die Folgeiteration. Sie entsprechen am Ende der Berechnung aller Iterationen meistens der Mittelung auf Basis der Alignierungsparameter, bei denen jedes Partikel genau einer Klasse zugeordnet ist.

4.2.2 Parallelisierungsstrategie

Für die performante Umsetzung des ML-Algorithmus lag der Schwerpunkt auf der Entwicklung einer skalierbaren Parallelisierungsstrategie mit den folgenden Eigenschaften:

1. Die Berechnung kann auf sehr viele parallele Prozesse (Rechnerknoten) verteilt werden (z.B. auf 16.000 Prozessoren des *Blue-Gene Computer-Cluster*)

2. Langsamere Prozessoren in einem *Computer-Cluster* mit inhomogener Infrastruktur (Rechner mit unterschiedlichen Prozessoren), dürfen die Gesamtgeschwindigkeit der Berechnung nicht ausbremsen.

3. Optimierter Speicherbedarf jedes Prozesses. Jeder Prozess muss alle Daten bekommen, die für die Berechnung notwendig sind, denn auf den Arbeitsspeicher eines anderen Prozesses kann bei MPI nicht zugegriffen werden (ein *Shared-Memory* Konzept bietet MPI nicht). Trotzdem müssen die verteilten Daten so gewählt sein, dass der eingeschränkte Arbeitsspeicher der Rechnerknoten ausreicht.

Um diese Eigenschaften zu erreichen, ist es zielführend, das Problem in möglichst viele unabhängige Teilprobleme zu zerlegen. Betrachtet man den ML-Algorithmus (Abschn.: 3.6), so könnte man intuitiv eine Aufteilung wählen, bei dem ein Prozess zur Berechnung der Metrik und der daraus resultieren neuen Modellparameter jeweils die Kombination aus einer Referenz und einem Partikel, zugeteilt wird. Bei einem Subtomogrammstapel von 1.000 Partikeln und einer Klassifikation mit zwei Klassen würden somit 2.000 Pakete zur Verteilung bereit stehen. Eine Aufteilung z.B. auf 16.000 Prozessoren des *Blue-Gene* wäre mit diesem Ansatz nicht mög-

lich. Dieses einfache Beispiel zeigt, das dieser Ansatz für die Zielplattform nicht ausreichend ist.

Um diese Beschränkung aufzulösen und gleichzeitig unabhängig von einer inhomogenen Infrastruktur zu sein, muss die Anzahl der zu verteilenden Aufgaben ein vielfaches zur Anzahl der verwendeten Prozessoren sein. Deshalb werden die Referenz-Partikel-Pakete zusätzlich über Blöcke aus der Liste der Rotationen verteilt. Dabei ist die Blockgröße beliebig einstellbar. Mit diesem Ansatz könnten z.B. 2.000 Referenz-Partikel-Pakete bei einer üblichen Liste von z.B. 10.000 Rotationen und einer Rotationsblockgröße von 500 in 40.000 Aufgaben aufgeteilt werden. Dabei ist natürlich zu beachten, dass der Aufwand für die Berechnung der Lösungsformeln des ML-Algorithmus im Verhältnis zum Rechenaufwand für die Verwaltung der Pakete und Aufgaben steht.

Die besondere Schwierigkeit ergibt sich für die frei skalierbare Verteilung beim ML-Algorithmus. Es ist nicht ausreichend, wie z.B. bei der Multi-Referenz-Klassifikation (Abschn.: 3.5), sich für jeden Vergleich von Partikel und Referenz einen optimalen Wert zu merken. Der ML-Algorithmus berechnet die neuen Referenzen für die Folgeiteration als gewichtete Mittelung der Partikel unter allen möglichen Alignierungen. Das bedeutet, die Mittelung wird unter Einbeziehung aller Orientierungen aus der Eingabe-Rotations-Liste und allen möglichen Verschiebungen in den drei Raumrichtungen berechnet. Der Wichtungsfaktor ergibt sich direkt aus der Posterioren Wahrscheinlichkeit $P(k, \phi | X_i, \Theta^{(t)})$ (Abschn.: 3.6, Gl.: 3.25), die die Grundlage für die Neuberechnung der Modellparameter (Gl.: 3.26, 3.27, 3.29, 3.30) beim ML-Algorithmus ist. Wesentlich ist, dass das Integral der Posterioren Wahrscheinlichkeiten (nach dem Bayestheorem) eins ergeben muss. Im Nenner dieser Funktion steht das Integral über die *Likelihood*-Funktion $\int P(X_i | k', \phi', \Theta^{(t)}) P(k', \phi' | \Theta^{(t)}) d\phi'$ als normierender Faktor. Damit wird deutlich, dass für die Berechnung des Integrals der Vergleich eines Partikel X_i mit allen Referenzen A notwendig ist.

Da die gesamte *Likelihood*-Funktion normalerweise nicht in den Speicher eines Rechnerknoten passt, müsste in einem naiven Ansatz zuerst das Integral über die ML-Funktion sukzessive bestimmt werden. Alle wertvollen Zwischenergebnisse würden während dieses ersten Laufes verworfen werden, denn der notwendige Speicherplatz zur Speicherung der Ergebnisse, wäre ein vielfaches der zu analysierenden Datenmenge der Subtomogramme. Die Neuberechnung der Parameter könnte dann erst in einen zweiten Lauf durchgeführt werden. Der größte Teil der Berechnungen wäre redundant.

Das Integral der Posterioren Wahrscheinlichkeit ist in allen Berechnungen ein konstanter Faktor und wirkt lediglich als Skalierung auf die einzelnen Ergebnisse. Allerdings lässt sich ohne die Skalierung die Mittelung innerhalb einer Klasse nicht durchführen. Um doppelte Berechnungen zu vermeiden und die Neuberechnung der Modellparameter in einem Schritt mit der Berechnung *Likelihood*-Funktion durchzuführen, merkt sich das Programm das lokale Teil-Integral (MLSUM) für jede berechnete Aufgabe. Sukzessive werden die Teil-Integrale während der Berechnung zusammengefasst und gleichzeitig die berechneten Modellparameter neu skaliert. Liegen alle Teil-Integrale für ein Partikel bzgl. aller Referenzen vor, hat das Partikel bzgl. jeder Referenz die richtige Skalierung bzw. den richtigen Wahrscheinlichkeitswert. Nun können Partikeln, die denselben Status erreicht haben, innerhalb einer Klasse (Referenz) gemittelt werden.

Außerdem wird im Laufe der Berechnung der *Likelihood*-Funktion jedes Volumen elementweise exponenziert. Mathematisch ist das möglich, Numerisch kommt der Computer aber an die Grenze der ausreichenden Genauigkeit der Gleitkommazahlen. Aus diesem Grund wird vor der Exponentierung, vom Volumen (V) das Maximum des Volumens ($max(V)$) abgezogen und als logarithmischer Skalierungsfaktor (LNSCALE) mitgeführt ($e^V = e^{V-max(V)} \cdot e^{max(V)}$).

LNSCALE ist ein zusätzlicher Skalierungsfaktor analog zu MLSUM. Während einer Iteration werden die beiden Faktoren für jedes Volumen bestimmt und bei der Berechnung der Klassenwahrscheinlichkeit und der Mittelung von Volumen als essentielle Faktoren eingerechnet.

4.2.3 Das Konzept der Pakete, Aufgaben und Befehle

Für die Parallelisierung wurde ein zentraler *Server*-Prozess implementiert, der vielen *Worker*-Prozessen Aufgaben zuteilt. Die *Worker* bearbeiten die Aufgaben in der Regel unabhängig voneinander. Nur in speziellen Fällen wird eine direkte Kommunikation zwischen den *Worker*-Prozessen aufgebaut.

Zu Beginn des Programms erstellt der *Server* eine Liste von Paketen, die er verwaltet. Jedes Paket referenziert ein Tupel aus einem Referenz-Volumen A_k und einem Partikel-Volumen X_i (zur Übersichtlichkeit wird eine vereinfachte Nomenklatur verwendet, die sich an vorherige Kapitel anlehnt). Die Anzahl der Pakete entspricht dem Produkt aus der Anzahl der Referenzen und der Anzahl der Partikel.

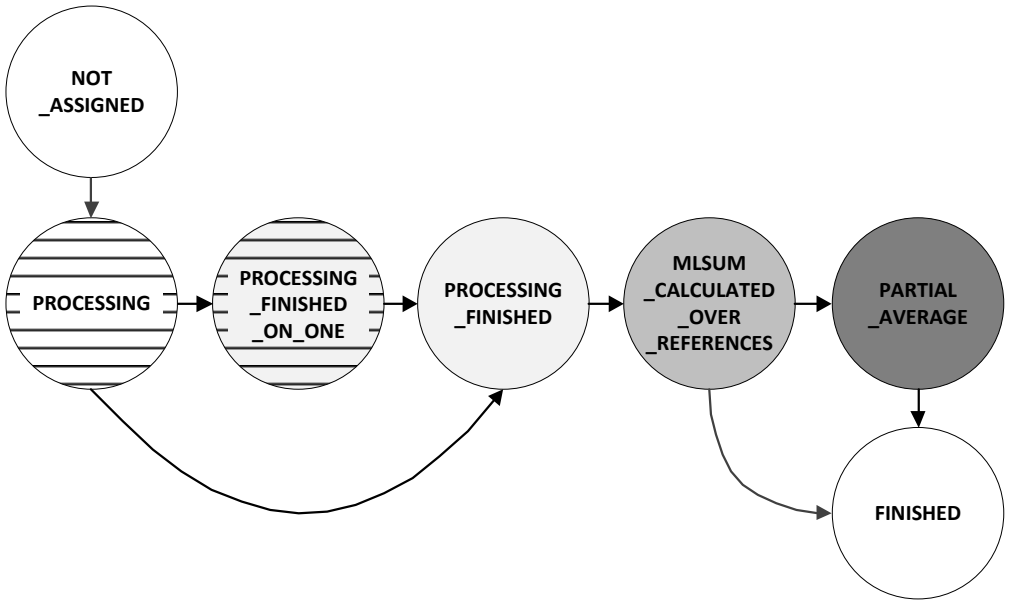


Abbildung 4.2: Zustandsdiagramm des Referenz-Partikel-Paketes. In der Parallelisierung muss ein Paket für die Berechnung fünf bis sieben aufeinander folgende Zustände durchlaufen. NOT_ASSIGNED - Paket wurde noch nicht verteilt. PROCESSING - Paket oder Teile des Paketes wurden an einen oder mehrere *Worker* verteilt und sind in Bearbeitung. PROCESSING_FINISHED_ON_ONE - Alle Teile des Paketes sind verteilt. Die Berechnung mindestens eines Teil-Paketes ist beendet und es ist noch mindestens ein Teil-Paket in der Berechnung. PROCESSING_FINISHED - Das gesamte Paket wurde berechnet. MLSUM_CALCULATED_OVER_REFERENCES - Alle Pakete innerhalb dieser Klasse und das Integral über die *Likelihood*-Funktion sind fertig berechnet. PARTIAL_AVERAGE - Die in dem Paket referenzierten Ergebnis-Volumen wurden mit den Ergebnis-Volumen mindestens eines anderen Paketes zusammen gemittelt. FINISHED - Das Paket ist leer oder enthält das vollständige Klassenmittel.

Die Pakete werden über die Rotationen in kleinere Teil-Pakete aufgeteilt. Dabei ist die gesamte Liste der Rotationen in gleichgroße Blöcke partitioniert, welche als Einheit dem *Worker* zugewiesen werden. Diese Blockgröße ist ein Konfigurationsparameter und soll so gewählt werden, dass die *Worker* eine geeignete Zeit beschäftigt sind, bevor sie sich wieder an den *Server* wenden.

Nach einem bestimmten Schema verteilt der *Server* die Teil-Pakete zur Berechnung des *Likelihood* und der neuen Modellparameter dynamisch an die *Worker*. Jedes Teil-Paket entspricht dabei einer Aufgabe.

Der *Server* selbst hält keine Volumen im Speicher sondern nur die Informationen über Pfade und Dateinamen. Diese Informationen bekommt der *Worker* über die Paketinformation. Die Volumen befinden sich auf dem gemeinsamen Dateisystem. Jeder *Worker* lädt sich seine benötigten Volumen von dort. Es werden keine Volumen vom *Server* an die *Worker* gesendet. In bestimmten Fällen sind die *Worker* allerdings in der Lage, Volumen untereinander auszutauschen.

Beim Programmstart liest der *Server*-Prozess die Konfiguration und verteilt sie an die *Worker*. Dadurch kennt jeder *Worker* die Dateinamen und Rotationslisten. Die Teil-Pakete sind einfach durchnummeriert. Damit reicht es, eine Ganzzahl an den *Worker* zu übertragen, um die Referenz, das Partikel und den Rotationsblock der Aufgabe zu bestimmen.

Die *Worker* kennen lediglich ihre eigenen Aufgaben und nur einen kleinen Ausschnitt der Daten (wenige Volumen). Nur der *Server* weiß, welcher der *Worker*-Prozesse welche Aufgaben bzw. Teil-Pakete bereits bekommen und ggf. berechnet hat, und welche Pakete noch verteilt werden müssen. Außer der Aufgabe zur Berechnung eines Teil-Paketes muss der *Server* noch weitere Befehle an die *Worker* übermitteln, damit ein Paket vollständig berechnet werden kann. Wann immer die *Worker*-Prozesse eine Aufgabe berechnet oder einen Befehl ausgeführt haben, melden sie sich beim *Server*, der ihnen dann eine neue Aufgabe oder einen neuen Befehl erteilt.

Damit ein Paket vollständig berechnet ist, muss es fünf bis sieben aufeinander folgende, fest definierte Zustände durchlaufen, die vom *Server* verwaltet und getriggert werden (Abb.: 4.2):

- **NOT_ASSIGNED**. Bisher wurde das Paket oder Teile des Paketes nicht an einen oder mehrere *Worker* verteilt.

- **PROCESSING**. Das Paket oder Teile des Paketes wurden an einen oder mehrere *Worker* verteilt und befinden sich in der Bearbeitung.

- **PROCESSING_FINISHED_ON_ONE**. Alle Teile des Paketes sind verteilt. Die Berechnung mindestens eines Teil-Paketes ist beendet und es ist noch mindestens ein Teil-Paket in der Berechnung.

- **PROCESSING_FINISHED**. Das gesamte Paket wurde berechnet, d.h. es wurden für dieses Paket die neuen **unskalierten** Modellparameter berechnet: das Integral über den zugeordneten Teil der *Likelihood*-Funktion (MLSUM) und der logarithmische Skalierungsfaktor (LNSCALE).

- **MLSUM_CALCULATED_OVER_REFERENCES**. Alle Pakete innerhalb dieser Klasse wurden fertig berechnet und das Integral über die *Likelihood*-Funktion wurde für dieses Paket (und damit auch für alle anderen Pakete der Klasse) bestimmt.

- **PARTIAL_AVERAGE**. Die in dem Paket referenzierten Ergebnis-Volumen wurden mindestens mit den Ergebnis-Volumen eines anderen Paketes zusammen gemittelt. Es fehlt aber mindestens noch ein Ergebnis-Volumen für das vollständige Klassenmittel.

- **FINISHED**. Das Paket hat entweder sein Ergebnis-Volumen zur Mittelung an ein anderes Paket abgegeben und ist leer, oder es hält selbst das vollständige Klassenmittel.

4.2.4 Der Programmablauf erklärt am Beispiel

Ein einfaches Beispiel soll Einblick in die wesentlichen Abläufe der Berechnung geben. Um die Komplexität gering zu halten, besteht der Partikelstapel aus drei Partikeln (X_1, X_2, X_3). Die Klassifikation wird mit drei Referenzen gestartet (A_1, A_2, A_3) und die Liste der Rotationen wurde in zwei Blöcke aufgeteilt (R_1, R_2). Damit verwaltet der *Server* neun Pakete. Jedes Paket kann in zwei Teil-Pakete zerlegt werden. Somit entstehen 18 Aufgaben, die im Beispiel von zwei *Worker*-Prozessen ($W1$, $W2$) verarbeitet werden.

Beim Start der Berechnung liest der *Server* die Konfiguration und erstellt die Liste mit neun Paketen und 18 Aufgaben. Alle Paket sind im Status

'NOT_ASSIGNED', da noch keine Aufgaben an die *Worker* verteilt wurden (Abb.: 4.3).

Der *Server* beginnt, die Aufgaben aus den einzelnen Paketen zu verteilen. Um die Kommunikation zwischen *Worker*-Prozessen zu minimieren und damit gleiche Volumen in der Regel nicht von mehreren *Worker*-Prozessen in den Speicher geladen werden müssen, versucht der *Server* Aufgaben eines Paketes immer an den gleichen *Worker* zu verteilen. Außerdem werden zuerst die Aufgaben zu allen Referenzen eines Partikels verteilt (spaltenweise in den Abbildungen), bevor Aufgaben zum nächsten Partikel in die Verteilung kommen. Es wird sich zeigen, dass durch dieses Vorgehen sich der Speicherplatzbedarf eines *Workers* häufig auf die Größe weniger Volumen reduziert.

Entsprechend wird in dem Beispiel eine Aufgabe aus dem Paket $A1X1$ an den *Worker* $W1$ und aus dem Paket $A2X1$ an den *Worker* $W2$ gegeben. Der Status beider Pakete geht auf 'PROCESSING' über. Die *Worker* laden das benötigte Referenz- und Partikel-Volumen und berechnen jetzt für die Partikel-Referenz-Kombination und den angegebenen Rotationsblock den entsprechenden Teil der *Likelihood*-Funktion und die neuen Modellparameter auf Basis der Posterioren Wahrscheinlichkeit (Abb.: 4.4). Nach der Berechnung sind die Ergebnis-Volumen 'falsch' skaliert. Die lokalen Skalierungs-Faktoren (MLSUM und LNSCALE) werden für die spätere Korrektur gespeichert.

Die *Worker* kehren nach der Berechnung der Aufgabe zum *Server* zurück und übergeben alle skalaren Modellparameter und die Skalierungsfaktoren. Der *Server* verteilt die übrigen Aufgaben der beiden Pakete und die *Worker* beginnen mit der Berechnung. Ein Laden von Referenz und Partikel ist nicht mehr nötig, da beides aus den Berechnung der vorherigen Aufgabe noch im Speicher liegt. Nach der Berechnung hält jeder *Worker* zwei (unterschiedlich skalierte) Ergebnisvolumen im Speicher (Abb.: 4.5).

Der *Worker* hält zwei Ergebnisvolumen des gleichen Paketes im Speicher. Ohne Aufforderung des *Servers* fasst der *Worker* die Volumen zu einem zusammen. Dabei wird LNSCALE verwendet, um die relative Skalierung der Ergebnisse zueinander zu berechnen und beide MLSUM-Werte werden addiert. Nachdem die *Worker* zum *Server* zurückgekehrt sind, wechselt der Status der ersten zwei Pakete auf 'PROCESSING_FINISHED'. Der *Server* verteilt zwei neue Aufgaben an die *Worker* entsprechend des Verteilungsschemas (Abb.: 4.6).

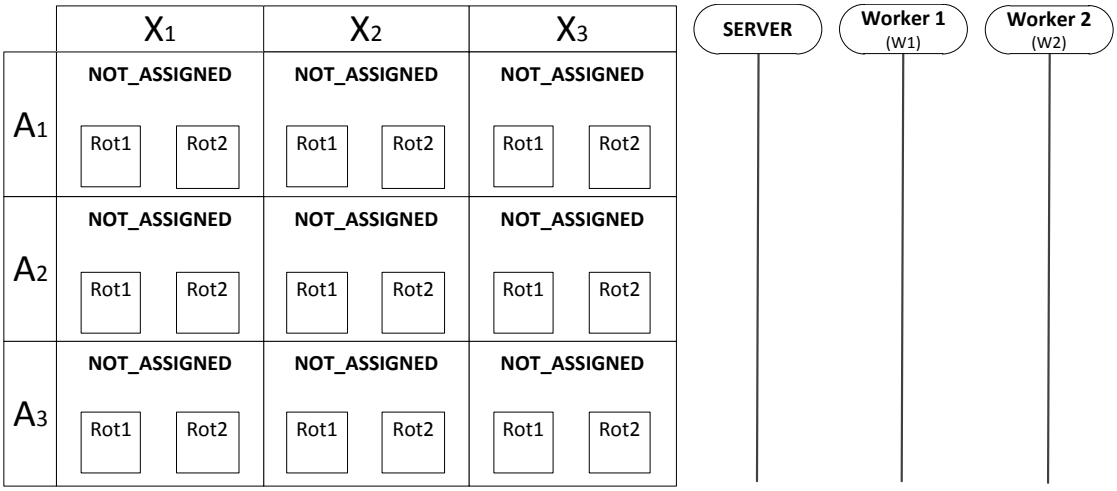


Abbildung 4.3: Paralleler ML-Algorithmus - Start-Konfiguration. Drei Partikel X_1 X_2 und X_3. Drei Klassen-Referenzen A_1, A_2 und A_3. Jede Klasse wird mit jedem Partikel kombiniert. Jede Kombination ergibt ein Paket. Zwei Rotationsblöcke R_1 und R_2 teilen jedes Paket in zwei Berechnungsaufgaben, die der *Server* auf zwei *Worker* verteilen kann. Zum Start der Berechnung ist noch kein Paket an einen *Worker* verteilt. Alle Pakete sind im Status 'NOT_ASSIGNED'.

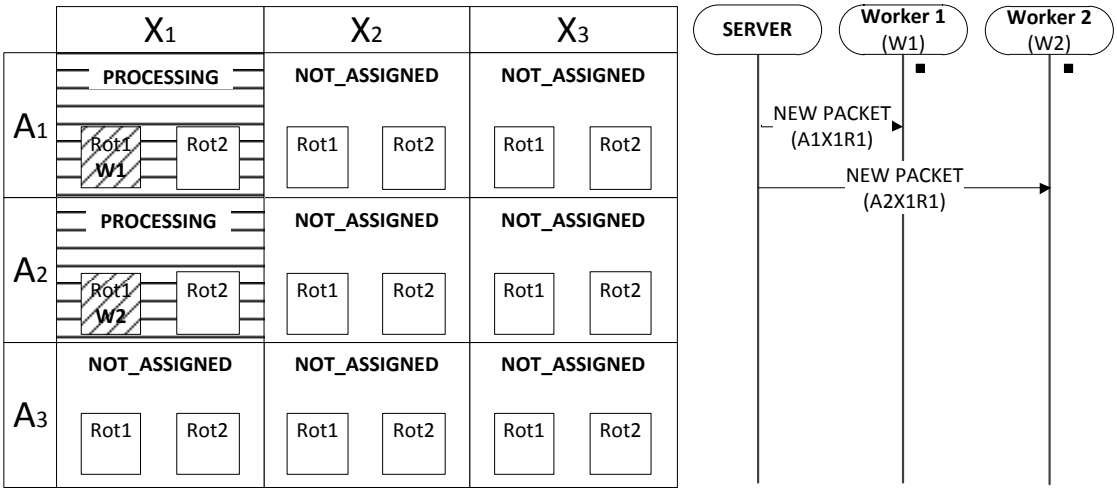


Abbildung 4.4: Paralleler ML-Algorithmus - Erste Aufgabenverteilung. Pakete, aus denen Aufgaben verteilt wurden, gehen in den Status 'PROCESSING'. Zur Minimierung des Speicherbedarfs und der Kommunikation zwischen den *Worker*, werden die Aufgaben spaltenweise verteilt und die Pakete dediziert einem *Worker* zugeordnet.

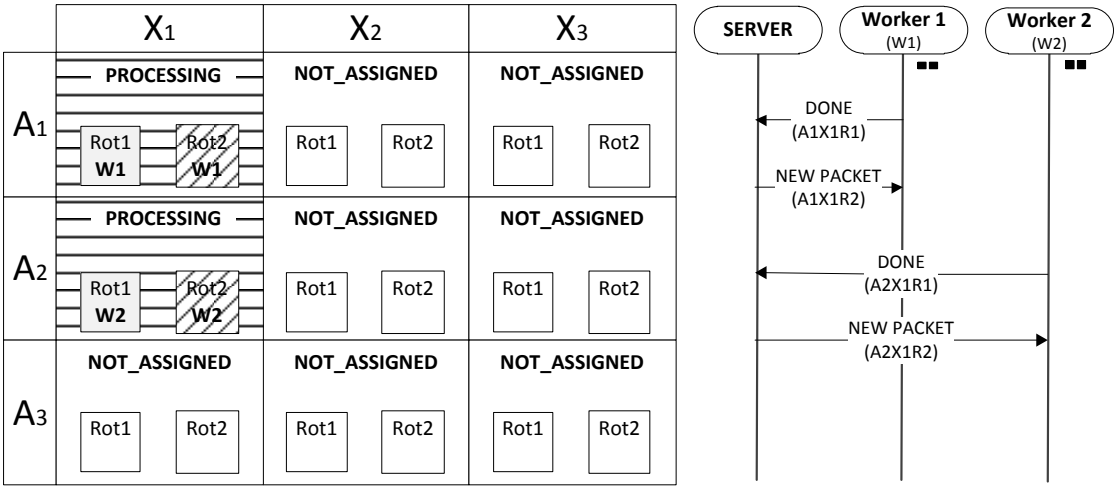


Abbildung 4.5: Paralleler ML-Algorithmus - Berechnung weiterer Aufgaben. Die *Worker* kehren zum *Server* zurück und erhalten sofort den zweiten Teil des Paketes zur Berechnung. Nach der Berechnung hält jeder *Worker* zwei Ergebnisvolumen im Speicher (kleine schwarze Quadrate).

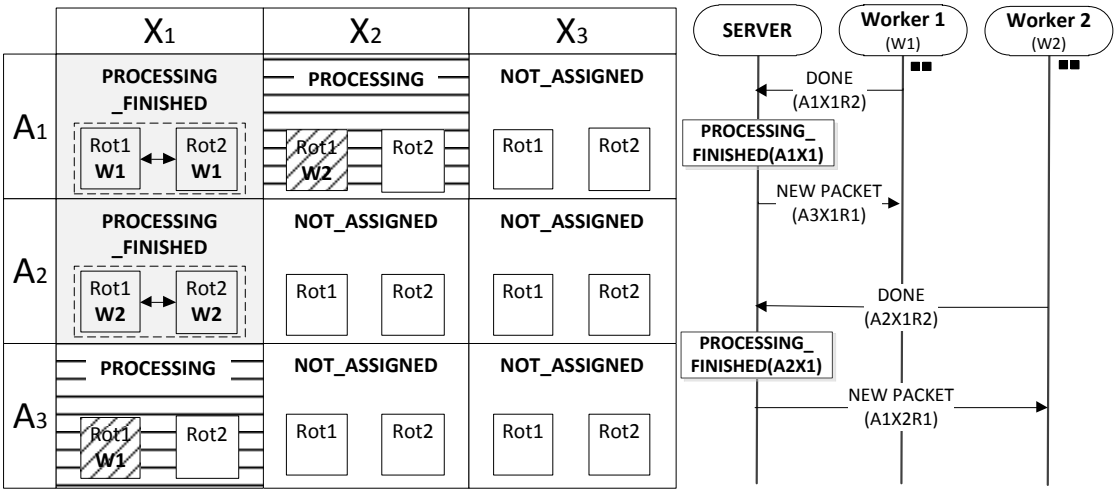


Abbildung 4.6: Paralleler ML-Algorithmus - Berechnung der ersten Pakete beendet. Die berechneten Pakete gehen in den Status 'PROCESSING_FINISHED' über und neue Aufgaben werden verteilt.

Die *Likelihood*-Funktion ist über ein Partikel mit allen Referenzen definiert (Abschn.: 3.6). Sind alle Pakete für ein Partikel mit allen Referenzen (Spalte) berechnet, so sind alle Teil-Integrale und damit das Intergral über die Posteriore Wahrscheinlichkeit für dieses Partikel ermittelt. Alle Ergebnisvolumen (innerhalb der Spalte) werden nun richtig skaliert und gehen in den Status ML-SUM_CALCULATED_OVER_REFERENCES (Abb.: 4.7).

Nachdem auch alle Pakete für das zweite Partikel (X_2) den Status ML-SUM_CALCULATED_OVER_REFERENCES erreicht haben, und damit richtig skaliert sind, können die Ergebnisvolumen innerhalb der Klassen gemittelt werden. Der *Server* ruft dafür den Befehl 'MERGE_PARTIAL_AVERAGE' auf dem *Worker*, der sich zuerst nach Erreichen dieses Status bei ihm meldet. Der *Server* teilt dem *Worker* mit, welche beiden Volumen zusammengefasst werden sollen, welcher *Worker* das andere Volumen hält. Hält der beauftragte *Worker* mehr Volumen im Speicher als sein Gegenüber, so wird der *Server* den *Worker* auffordern, sein Volumen abzugeben. Ist es anders herum, empfängt der beauftragte *Worker* das Ergebnisvolumen. Damit wird der benötigte Speicherplatz automatisch gleichmäßig auf die *Worker* verteilt. Das 'MERGE_PARTIAL_AVERAGE' hat immer Vorrang vor dem Verteilen neuer Aufgaben. Dadurch wird sichergestellt, das der benötigte Speicherplatz minimal gehalten wird.

Der *Worker*, der diesen Befehl erhält, kommuniziert direkt mit dem Ziel-*Worker*, der das Volumen hält und ruft den Befehl 'MERGE_PA' am Ziel-Woker auf. Auch während der Bearbeitung einer Aufgabe prüft jeder *Worker* regelmäßig, ob er durch einen anderen Befehl unterbrochen wurde. In diesem Fall stoppt der Ziel-*Worker* die aktuelle Aufgabe, erfüllt den 'MERGE_PA' Befehl des anderen *Worker* und führt danach seine Arbeit an der Aufgabe fort (Abb.: 4.8).

Wird der *Server* über den erfolgreich abgeschlossenen MERGE Befehl informiert, so bekommt das Paket, das auf gemittelte Volumen verweist, den Status PARTIAL_AVERAGE. Das andere Paket verweist auf kein Ergebnisvolumen mehr und ist leer. Es wird auf den Status 'FINISHED' gesetzt (Abb.: 4.9).

Die beiden Aufgaben des letzten Paketes werden auf zwei unterschiedliche *Worker* verteilt. Die Situation tritt immer genau dann ein, wenn bereits alle Pakete zugeordnet sind und ein *Worker* nach einer weiteren Aufgabe fragt. Im Beispiel kehrt der *Worker* Eins als erstes zum *Server* zurück und meldet, dass er seine Aufgabe erledigt hat. Das Paket geht in den Zustand 'PROCESSING_FINISHED_ON_ONE', da alle Aufgaben des Paketes verteilt sind, die

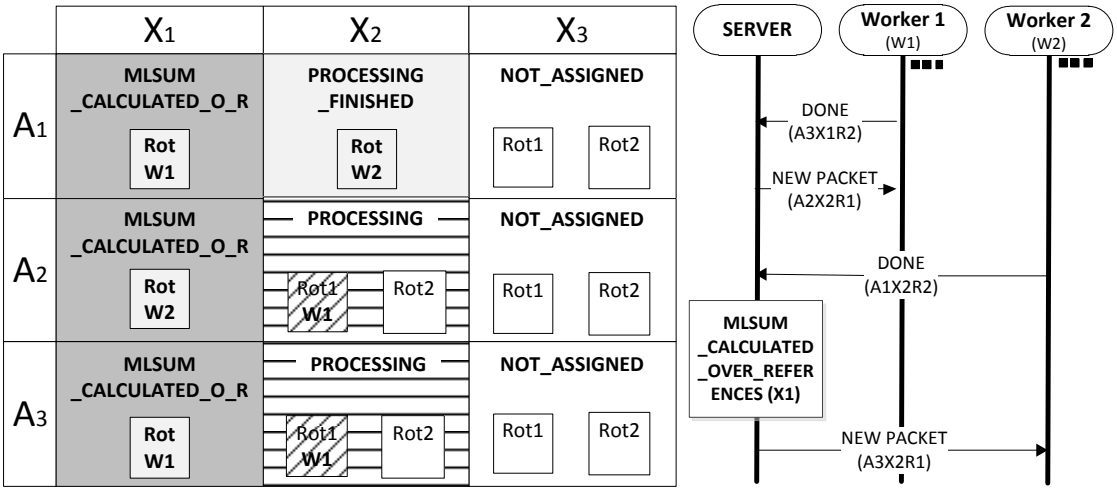


Abbildung 4.7: Paralleler ML-Algorithmus -Intergral der *Likelihood*-Funktion für ein Partikel ermittelt. Die Ergebnisvolumen bzgl. des Partikel X_1 sind nun richtig skaliert.

Berechnung mindestens eines Teil-Paketes beendet ist und noch mindestens eine Aufgabe des Paketes in der Berechnung ist.

Da keine weiteren Aufgaben zur Berechnung anstehen und kein anderer *Worker* ein berechnetes Ergebnisvolumen zum Mitteln hält, wird der *Worker* Eins aufgefordert, auf die nächste Mittelung zu warten. Der *Server* sendet den Befehl WAIT_FOR_PASSIVE_MERGE zum *Worker* Eins (Abb.: 4.10).

Worker Eins bekommt das Ergebnisvolumen zum zweiten Rotationsblock vom Paket A_3X_3 und fasst es mit dem Volumen des ersten Rotationsblocks zusammen. Damit ist die letzte Aufgabe aus dem letzten Paket berechnet. Gleich danach liegt auch das Integral der ML-Funktion für das letzte Partikel (X_3) vor. Die finale Mittelung über alle Partikel einer Klasse kann nun abgeschlossen werden.

Der *Server* fordert den *Worker* Zwei auf, das Ergebnisvolumen für den zweiten Rotationsblock mit *Worker* Eins auszutauschen (er hält das Ergebnisvolumen für den ersten Rotationsblock). Der *Server* sendet den Befehl MERGE_PROCESSING für das Paket der ersten Klasse (A_1) an *Worker* Eins. Der *Worker* Eins fordert der *Worker* Zwei auf, ihm das Ergebnisvolumen für die Mittelung zu senden. Alle Ergebnisvolumen für die Klasse Zwei (A_2) werden von *Worker* Zwei und alle Ergebnisvolumen von Klasse Drei (A_3) werden vom *Worker* Eins gehalten. Ohne weitere Aufforderung vom *Server* fassen die *Worker* diese Volumen intern zusammen (Abb.: 4.11).

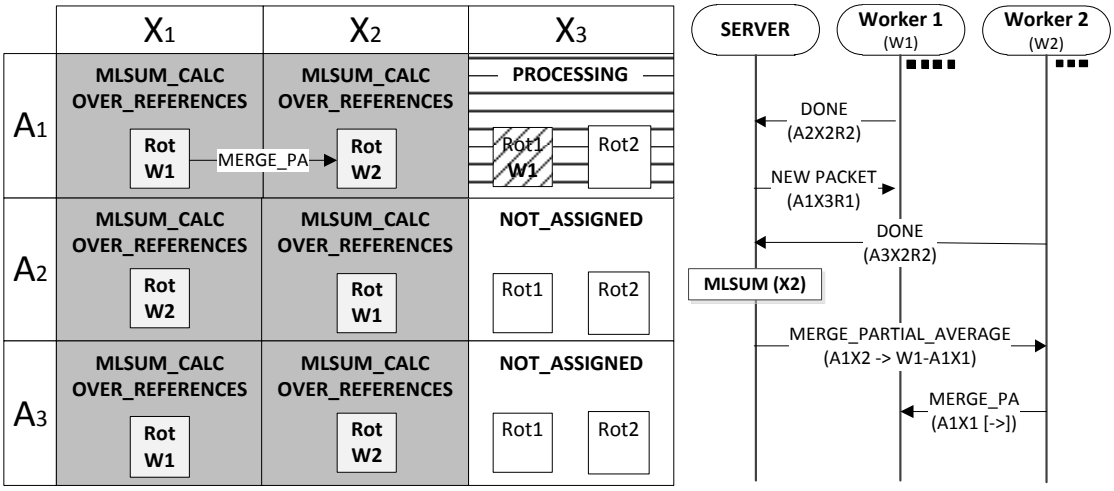


Abbildung 4.8: Paralleler ML-Algorithmus - *Server*-Befehl zur Mittelung von Ergebnisvolumen. Alle Pakete für Partikel X_1 und X_2 haben den Status MLSUM_CALCULATED_OVER_REFERENCE erreicht und sind damit richtig skaliert. Der *Server* fordert den nächsten *Worker*, der sich meldet und eines dieser Pakete hält, auf, das bestimmte Ergebnisvolumen mit einem anderen *Worker* zu mitteln.

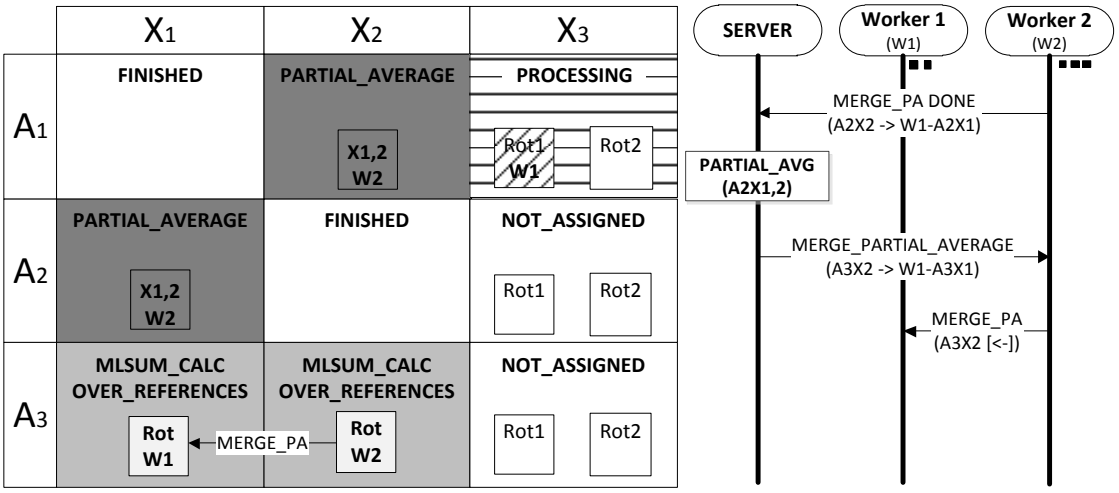


Abbildung 4.9: Paralleler ML-Algorithmus - Partielle Mittelung von Ergebnisvolumen. Pakete, die einen Teil einer Mittelung referenzieren, gehen in den Status PARTIAL_AVERAGE. Pakete, die kein Ergebnisvolumen mehr referenzieren, gehen in den Status 'FINISHED'.

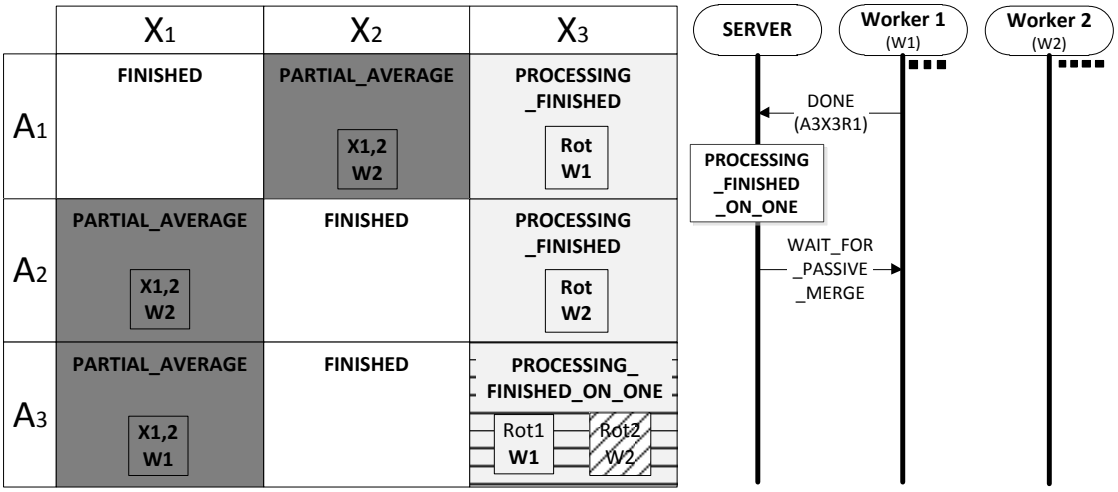


Abbildung 4.10: Paralleler ML-Algorithmus - Zwei *Worker* bearbeiten ein Paket. Alle Pakete sind zugeordnet und *Worker* Zwei hat eine Aufgabe aus dem letzten Paket erhalten, an dem schon *Worker* Eins arbeitet. *Worker* Eins hat seine Augabe beendet, das Paket geht in den Zustand 'PROCESSING_FINISHED_ON_ONE'. Es gibt keine weiteren Aufgaben oder Berechnungen. *Worker* Eins muss darauf warten, sein Ergenis-Volumen abzugeben oder ein Volumen zur Mittelung zu erhalten.

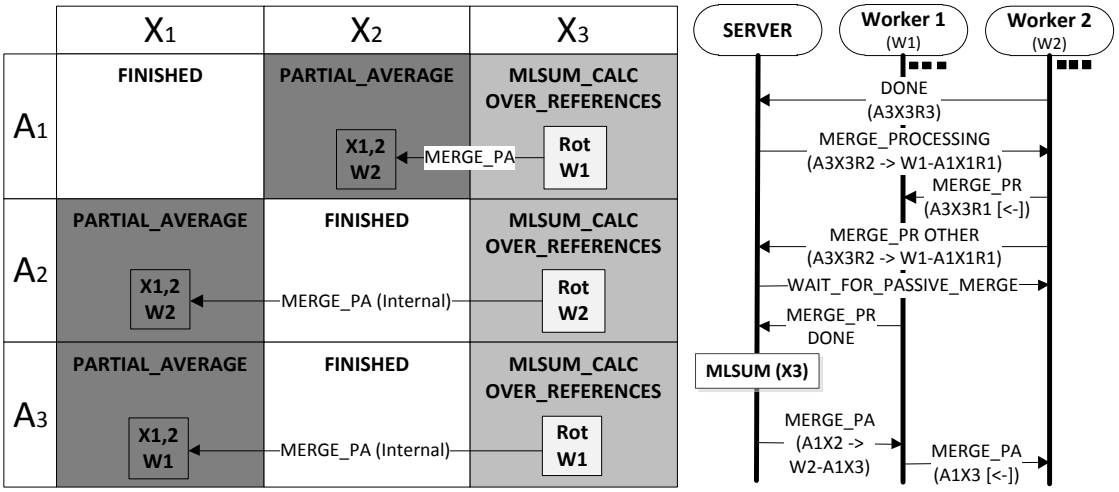


Abbildung 4.11: Paralleler ML-Algorithmus - Abschließende Mittelung innerhalb der Klassen. Nachdem alle Pakete zum Partikel X_3 im Status 'MLSUM_CALCULATED_OVER_REFERENCES' stehen, fordert der *Server* die *Worker* auf, die Ergebnisvolumen der Klasse Eins zu mitteln. Für Klasse Zwei und Drei wird die Mittelung selbständig ohne weiteren Befehl von den *Worker*-Prozessen durchgeführt, da alle Ergebnisvolumen einer Klasse bei genau bei einem *Worker* liegen.

Die *Worker* kehren zum *Server* zurück und bestätigen die Ausführung der letzten Mittelungen. Damit ist das Ende der Iteration erreicht. Die Ergebnisse von unterschiedlichen *Worker*-Prozessen liegen im Speicher. Der *Server* fordert alle *Worker* auf, die Iteration zu beenden. Die *Worker* schreiben daraufhin ihre Ergebnisse auf die Festplatte und eine neue Iteration kann beginnen (Abb.: 4.12).

4.2.5 Bewertung der Implementierung

Der Schwerpunkt bei der parallelisierten Implementierung des ML-Algorithmus lag auf der Entwicklung eines frei skalierbaren Programms, das die Berechnung auf sehr viele parallele Prozesse (Rechnerknoten) in einer inhomogenen Infrastruktur verteilen kann. Gleichzeitig versucht das Programm, den Speicherbedarf pro Rechnerknoten zu minimieren. Der Grad der Parallelisierung der gewählten Implementierung ist nicht beliebig.

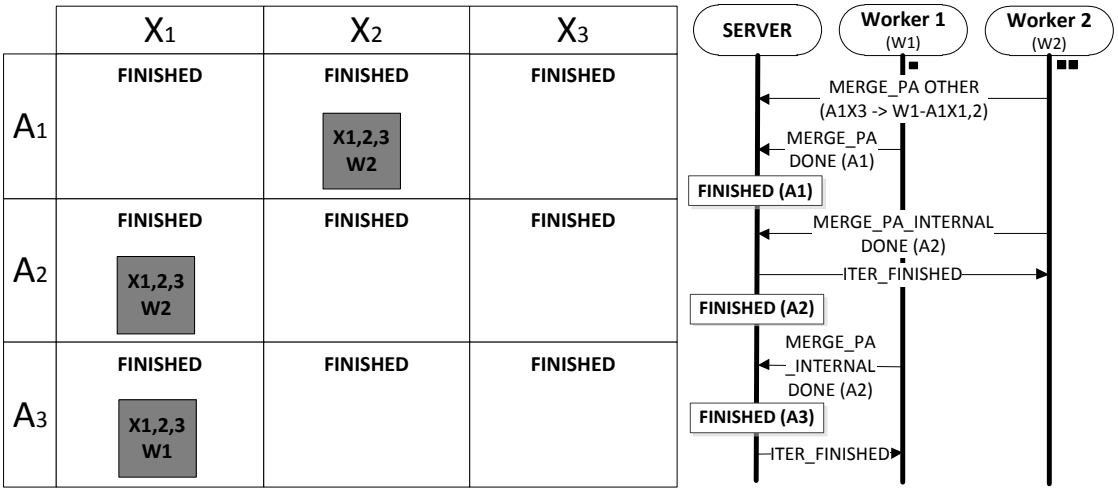


Abbildung 4.12: Paralleler ML-Algorithmus - Abschluss der Iteration. Alle Pakete sind im Status 'FINISHED'. Die Ergebnisvolumen liegen im Speicher unterschiedlicher *Worker*. *Worker* Eins hält ein Volumen und *Worker* Zwei hält zwei Volumen. Der *Server* fordert die *Worker* auf, die Iteration zu beenden und die Ergebnisse zu speichern.

Die Wartezeit zwischen den Iterationen ist abhängig vom Status aller Pakete. Erst, nachdem alle Pakete berechnet und die Ergebnisse zusammengefasst wurden, stehen die Klassenmittel und Eingangsparameter der Folgeiteration fest und die nächste Iteration kann begonnen werden. Die Wartezeit eines Workers beginnt in dem Moment, in dem er auf Nachfrage beim *Server* keine Aufgaben zur Bearbeitung mehr erhält. Die Wartezeit endet, wenn alle *Worker* die noch ausstehenden Pakete berechnet haben. Damit steigt die mögliche Wartezeit am Ende jeder Iteration mit der Größe der Pakete. Dies gilt insbesondere, wenn ein *Computer-Cluster* aus langsamen und schnellen Prozessoren besteht.

Umgekehrt können viele kleine Pakete zu einem hohen Kommunikationsaufkommen zwischen *Server* und den *Worker*-Prozessen führen und den *Server* überlasten. Wenn viele *Worker* zugleich neue Aufgaben anfordern oder der *Server* mit der Verteilung der Pakete nicht nachkommt, werden *Worker* blockiert.

Beide Faktoren können die Effizienz einschränken. Die Anzahl bzw. Größe der Pakete muss vor der Iteration, abgestimmt auf die Architektur des *Computer-Clusters*, konfiguriert werden. In der Regel dauert dann die Berechnung einer Iteration wesentlich länger als die Berechnung eines Pakets. Die Wartezeit am Ende der Iteration ist dann vernachlässigbar.

Wartezeiten können auch durch die Kommunikation zwischen den *Worker*-Prozessen entstehen. Nachdem ein *Worker* seine Aufgabe berechnet hat, müssen die Ergebnisse mit den Ergebnissen anderer *Worker* kombiniert werden. Diese Ergebnisse sind Volumen und Skalierungsfaktoren im Hauptspeicher des Workers. Kombinieren bedeutet hier, dass ein *Worker* das Ergebnis eines anderen Workers bekommt und zusammenfasst.

Beim Zusammenfassen wird eine gewichtete Summe der Ergebnisse berechnet. Somit steigt der Speicherbedarf durch das Kombinieren nicht. Jeder *Worker* hält zu jedem Zeitpunkt maximal so viele Ergebnisse im Speicher wie die Anzahl an Referenzen. Kann ein *Worker* zu einem Zeitpunkt seine Ergebnisse mit niemandem kombinieren, wird er seine bisherigen Ergebnisse vorhalten, bis der *Server* oder ein anderer *Worker* ihn informiert, dass seine Ergebnisse kombiniert werden können.

Zum Abgleich versucht der Worker eine MPI-Nachricht an seinen Partner zu senden. Der andere *Worker* ist sehr wahrscheinlich gerade dabei, eine andere Aufgabe zu berechnen. Daher prüft jeder *Worker* in regelmäßigen Abständen, ob ein anderer *Worker* versucht, seine Ergebnisse mit ihm zu kombinieren. Ist dies der Fall, unterbricht der *Worker* seine Arbeit und empfängt oder sendet sein zu kombinierendes Ergebnis. Der *Server* entscheidet, welcher der teilnehmenden *Worker* sein Ergebnis abgibt, und welcher ein Ergebnis aufnimmt und kombiniert. Der *Worker*, der das Ergebnis empfangen hat, kombiniert es mit seinem Ergebnis und beide *Worker* setzen ihre Arbeit fort.

Tatsächlich müssen Ergebnisse relativ selten kombiniert werden. Denn der *Server* versucht die Aufgaben so zu verteilen, dass jeder *Worker* seine eigenen Ergebnisse prozesslokal kombinieren kann. So wird der *Server* zuerst versuchen, jedem *Worker* ein eigenes Referenz-Partikel-Paar bzw. ein eigenes Paket zuzuweisen. Nach und nach versucht der *Server*, dem *Worker* alle Rotationsblöcke des Paketes zu zuteilen. Diese kann der *Worker* direkt bearbeiten und kombinieren, ohne die Ergebnisse eines anderen Workers zu benötigen. Zusätzlich entfällt das Laden der Eingabe-Volumen, da diese noch von der Berechnung des vorhergehenden Rotationsblocks im Speicher des Workers liegen.

Eine Situation, in der alle *Worker* versuchen, ihre Ergebnisse an einen bestimmten Prozess zu schicken, könnte schnell zu einem Flaschenhals werden und ebenfalls die Wartezeiten beteiligter Prozesse in die Höhe treiben. Aus diesem Grund achtet der *Server* darauf, dass alle *Worker* in ausgeglichener Anzahl Ergebnisse empfangen und senden, sodass dieses Problem vermieden wird.

Bei der Implementierung werden in jeder Iteration die Leerlaufzeiten bzw. die Wartezeiten der *Worker* protokolliert und können leicht ausgewertet werden. Bei Analysen stellte sich heraus, dass trotz der hier genannten auftretenden Einschränkungen die Leerlaufzeiten bei einer sinnvollen Konfiguration (insbesondere der Größe der Pakete), unter 1% liegen. Ausserdem lässt sich feststellen, dass nur selten eine redundante Berechnung der Ergebnisse notwendig ist.

Eine mögliche Verbesserung könnte durch den Einsatz mehrerer *Server*-Prozesse erreicht werden. Dafür würde man die *Worker* in mehrere Rechengruppen aufteilen und jeweils einem *Server* zuordnen. Für die bisher berechneten Datenmengen und Clusterarchitekturen ist die Skalierung der Implementierung ausreichend.

5 Ergebnisse

Um die Effektivität der vorgestellten Metriken und Algorithmen zu testen, wurden unterschiedliche Versuche mit simulierten Daten durchgeführt. Im Vergleich zu anderen Analyseverfahren zeigt MLTOMO bezogen auf die Unabhängigkeit von initialen Referenzen und die Qualität der Alignierung und Klassifikation eine deutliche Verbesserung. Im zweiten Abschnitt werden Analysen von experimentellen Daten gezeigt, die mit MLTOMO durchgeführt wurden. Bei den Daten handelt es sich um Subtomogramme aus kryo-elektronentomographischen Aufnahmen von Proben aufgereinigter Proteinkomplexe und ganzen Zellen. Das Signal-Rausch-Verhältnis war in allen Aufnahmen gering. Die Analysen wurden mit geringem A-Priori Wissen gestartet, um die Beeinflussung der Ergebnisse zu minimieren.

5.1 Vergleiche der Verfahren auf Basis simulierter Daten

Für die Erzeugung der simulierten Daten wurden zum Teil aus bekannten Röntgenstrukturen oder auch Einzelpartikel-Analysen makromolekulare Komplexe als Grundlage verwendet. In anderen Versuchen wurden primitive 3D-Referenzen eingesetzt. Um einen Test-Datensatz zu erzeugen, der möglichst nahe an experimentelle Aufnahmen herankommt, wurden die Daten mit unterschiedlich starkem Rauschen überlagert und der *Missing Wedge* wurde künstlich induziert. Die Lage und Orientierung der Subtomogramme wurde anhand von gemessenen Verteilungen in experimentellen Datensätzen eingestellt. Die Parameter wie Position und Orientierung der Partikel, das Signal-Rausch-Verhältnis, Größe und Masse der Partikel, die Anzahl der Partikelklassen und die Größe des fehlenden Kippwinkelbereiches waren bei den simulierten Daten A-Priori bekannt und lassen es zu, die Qualität des untersuchten Analyseverfahrens genau zu bestimmen. Für die Erzeugung der Phantom-Datensätze wurde die TOM-Toolbox verwendet [Nickell et al., 2005].

5.1.1 Vergleich von Maximum CCF Alignierung und MLTOMO

Für den Vergleich der Maximum CCF Alignierung (Abschn.: 3.4) und MLTOMO (Abschn.: 3.6), wurde ein Phantom-Subtomogramm-Datensatz generiert. Die Referenzstruktur war das Thermosom (molekulare Masse, 900 kDa) als großer molekularer Komplex. Es besteht aus zwei gestapelten Ringen. Die Ringe bilden einen Torus mit einer innen liegenden Kammer. Das Thermosom unterliegt großen Konformationsänderungen in seiner biologischen Umgebung. Es kann in einem geschlossenen und offenen Zustand vorliegen. Für die simulierten Subtomogramme wurde die offene Konformation gewählt, für die keine Kristallstruktur vorliegt. Aus diesem Grund war hier eine pseudo-atomare Struktur, die durch einen hybriden Ansatz ermittelt wurde [Nitsch et al., 1998], die Basis für den Datensatz. Damit sind Partikel in Größe, Form und Symmetrie vergleichbar mit der offenen Konformation des makromolekularen Thermosom-Komplexes. Im Detail wurde der Datensatz wie folgt erstellt:

1. Die Thermosom-Referenzstruktur wurde tiefpassgefiltert bis zu einer Auflösung von 4.5 nm. Bei dieser Auflösung lässt sich die achtfache Symmetrie noch deutlich erkennen. Die Voxelgröße wurde auf 1.5 nm Kantenlänge gesetzt und das Subvolumen besteht aus 28x28x28 Voxel.

2. Es wurden 300 Kopien des Referenzvolumens erzeugt. Die Partikel in den Volumen wurden zufällig translatiert und rotiert mit zuvor definierten anisotrop verteilten Rotationssatz. Dieser Satz an Rotationen simuliert eine bevorzugte Orientierung der Partikel in der Eisschicht der Probe.

3. Zu jedem der 300 Subvolumen wurde ein zufällig erzeugtes Volumen mit Gaußschem Rauschen addiert. Die Amplitude des Rauschens war so skaliert, dass das Signal-Rauschverhältnis bei 0.04 lag.

4. Für jedes Subvolumen wurde eine tomographische Kippserie von -60° bis +60° simuliert, indem die rotierten Phantom-Strukturen mit einem passenden *Missing Wedge* Volumen gefaltet wurden.

Die 300 simulierten Thermosom-Subtomogramme wurden mit MLTOMO und Maximum CCF aligniert. Um die Ergebnisse für die Maximum CCF Alignierung zu berechnen, wurde MLTOMO verwendet und Sigma auf einen sehr kleinen konstanten Wert gesetzt. Dabei wird ausgenutzt, dass der *Maximum Likelihood* Algorithmus, für sehr niedrige Simga Werte, äquivalent zum Maximum CCF Algorithmus

ist [Sigworth, 1998]. In diesem Fall werden die Wahrscheinlichkeitsdichtefunktionen zu einem Maximum. Beide Verfahren benutzten die *Compound Wedge* Metrik (Gl.: 3.13) und wurden mit einer identischen rotationsgemittelten Referenz aller Subtomogramme gestartet. Als Maß für die Qualität der Alignierung wurde der *Likelihood* (Gl.: 3.16) nach jeder Iteration berechnet (Abb.: 5.1 g). Es konnte festgestellt werden, dass der korrelationsbasierte Ansatz bereits nach wenigen Iterationen in einem lokalen Minimum konvergiert (Abb.: 5.1 f). Denn das berechnete 3D-Modell ist tonnenförmig, zeigt aber keine Symmetrie. MLTOMO zeigt ein langsameres Konvergenzverhalten aber konvergiert zu einer Struktur, die einen höheren *Likelihood* Wert erreicht. Das 3D-Modell zeigt die richtige, typisch achtfache symmetrische Struktur des Thermosoms (Abb.: 5.1 c).

5.1.2 Vergleich von Einschritt- und Zweischrittverfahren

Viele Ansätze zur Analyse von Subtomogrammen folgen einem Zweischrittverfahren, bestehend aus einer initialen Alignierung und gefolgt von einer Klassifikation der Subtomogramme. Dabei wird nach der initialen Alignierung auf eine Referenzstruktur, die strukturelle Varianz der Subtomogramme untereinander mittels Klassifikation untersucht. Im Gegensatz dazu folgt die Multi-Referenz-Klassifikation (Absch.: 3.5) einem Einschrittverfahren, indem die Alignierung und Klassifikation zugleich berechnet wird. In diesem Versuch werden beide Ansätze anhand eines konstruierten Datensatzes verglichen.

Grundlage für den Datensatz sind zwei primitive 3D-Referenz-Strukturen in einem 32x32x32 voxelgroßen Volumen. Die erste Struktur ist eine Hantel (Abb.: 5.2 a) und die zweite eine Kugel (Abb.: 5.2 b). Für den Datensatz wurden 380 Kopien der Hantelstruktur und 20 Kopien der Kugel erzeugt. Die Bilder in den Volumen wurden zufällig translatiert und rotiert. Zu jedem Volumen wurde ein zufällig erzeugtes Volumen mit Gaußschem Rauschen addiert. Die Amplitude des Rauschens war so skaliert, dass das Signal-Rauschverhältnis bei 0.06 lag.

Die Alignierung und Klassifikation wurde zum einen mit MLTOMO durchgeführt und zum anderen wurde der Datensatz im ersten Schritt aligniert (Abschn.: 3.4) und in einem zweiten Schritt mit PCA und *K-Means Clusterung* klassifiziert. Für das Zweischrittverfahren wurde das EM Softwarepaket von Hegerl verwendet [Hegerl, 1996]. Die Klassifikation wurde für $K = 2$ und $K = 3$, mit K als Parameter für die Anzahl der Klassen, durchgeführt. Für den *PCA*-basierten Ansatz und $K = 2$ wurden 387 Volumen zur ersten Klasse und nur 13 Volumen zur zweiten Klasse zugeordnet. 7 Kugelstrukturen wurden falsch klassifiziert. Mit $K = 3$

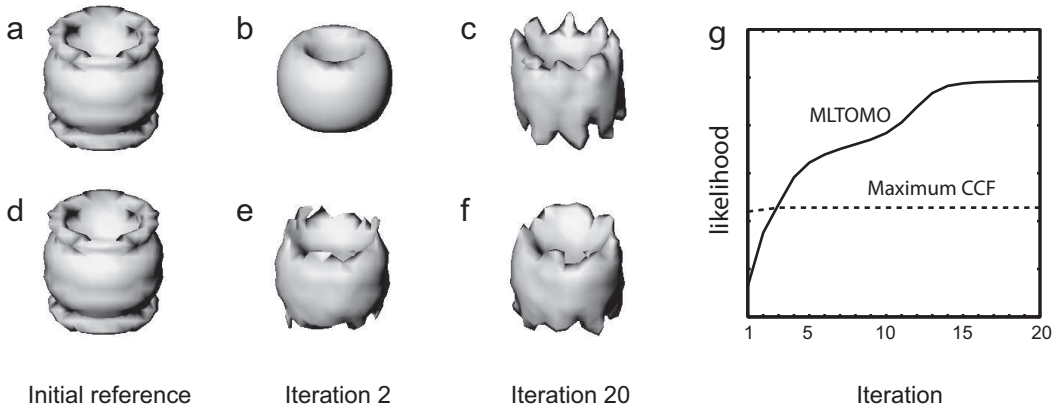
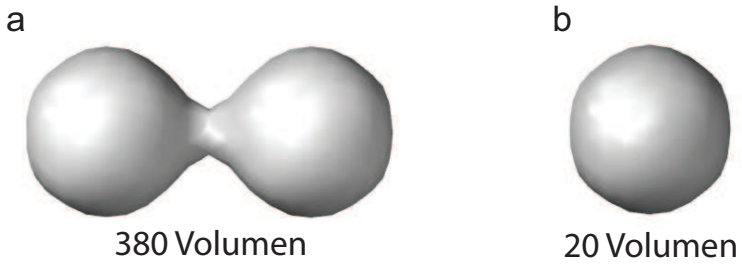


Abbildung 5.1: Vergleich von Maximum CCF Alignierung und MLTOMO. Alignierungsschritte von simulierten Thermosom-Subtomogrammen (N=300, Kippwinkelbereich: -60° to +60°, Singal-Rausch-Verhältnnis=0.04) unter Verwendung des *Maximum Likelihood* (a-c) und des korrelationsbasierten Ansatzes (d-f). Beide Algorithmen wurden mit einer identischen Referenz initialisiert(a,d). Der korrelationsbasierte Ansatz konvergiert nach wenigen Iterationen (e) in einem lokalen Minimum (f). MLTOMO zeigt eine geringere Auflösung zu Beginn (b), aber findet das globale Maximum (Iteration 20) und damit die richtigen Alignierungsparameter. Die *Likelihood* Funktion (g) beschreibt das Konvergenzverhalten von MLTOMO (durchgezogene Linie) und des korrelationsbasierten Ansatzes (gestrichelte Linie). Die Maximum CCF Alignierung erreicht schnell ein Plateau, aber ist im lokalen Maximum gefangen. MLTOMO zeigt ein langsameres Konvergenzverhalten. Der *Likelihood* steigt stetig und erreicht am Ende einen höheren Gesamtwert. Die Schwellwerte für die Oberflächendarstellung wurden auf eine molekulare Masse von 900 kDa eingestellt.



Methode	Alignierung & Klassifikation	Anzahl der Klassen (K)	Klasse 1	Klasse 2	Klasse 3	Fehler
PCA & K-Means Clustering	In getrenten Schritten	K=2	387	13	-	Sieben falsch klassifizierte Subtomogramme
		K=3	380	13	7	
MLTOMO	In einem Schritt	K=2	380	20	-	Kein Fehler
		K=3	380	20	0	

Abbildung 5.2: Vergleich von Alignierung und Klassifikation in einem oder in getrennten Schritten. Untersucht wurden zwei Verfahren (PCA-basierte Klassifikation und MLTOMO) anhand eines konstruierten Datensatzes mit 380 Hantelstrukturen (a) und 20 Kugelstrukturen (b). Die Tabelle zeigt die Klassifikationsergebnisse für beide Verfahren und eine unterschiedliche initiale Anzahl an Klassen K. Beim Zweischrittverfahren (initiale Alignierung gefolgt von PCA und *K-Means Clustering*) werden sieben Strukturen falsch zugeordnet. MLTOMO liefert für $K = 2$ und $K = 3$ das richtige Ergebnis.

werden die 380 Hantelstrukturen beim Zweitschrittverfahren zwar der richtigen Klasse zugeordnet, allerdings befanden sich wieder nur 13 Kugeln in der Klasse zwei und dafür sieben Kugeln in der Klasse drei. Bei genauerer Analyse der Daten wird deutlich, dass bereits die Alignierung der Strukturen fehlschlägt und die Klassifikation damit kein valides Ergebnis mehr erzeugen kann. Eine solche Fehlalignierung ist immer dann sehr wahrscheinlich, wenn sich die Partikel innerhalb eines Datensatzes stark in Struktur und Größe unterscheiden. Die Klassifikation mit MLTOMO führte für $K = 2$ und $K = 3$ zum richtigen Ergebnis und zeigte damit ein robusteres Verhalten.

5.1.3 Vergleich von unterschiedlichen Klassifikationsverfahren

Um die verschiedenen Klassifikationsverfahren zu testen, wurde ein Datensatz mit 200 Thermosom- und 200 Proteasom-Subtomogrammen mit 20 verschiedenen Signal-Rausch Verhältnissen, in einem Wertebereich von 0.0025 bis 1.0, erzeugt. Beide Partikel haben eine ähnliche Form und molekulares Gewicht. Wie bereits zuvor war die pseudo-atomare Struktur von Nitsch [Nitsch et al., 1998] die Basis für die Referenzstruktur des Thermosoms (molekulare Masse, 900 kDa) (Absch.: 5.1.1). Die Referenzstruktur des 20S Proteasom wurde auf Basis der atomaren Struktur aus der PDB erstellt (1PMA). Beide Strukturen bestehen aus zwei gestapelten Ringen. Die Ringe bilden einen Torus mit einer innen liegenden Kammer. Im Detail wurde der Datensatz wie folgt erstellt:

1. Die Referenzstrukturen wurden mit einem Tiefpassfilter auf die Auflösung von 4.5 nm justiert. Die Voxelgröße wurde auf 1.5 nm Kantenlänge gesetzt und das Subvolumen besteht aus 28x28x28 Voxel. Bei dieser Auflösung lässt sich die siebenfache Symmetrie des Proteasoms von der achtfachen Symmetrie der Thermosomen noch deutlich unterscheiden.

2. Für eine simulierte Kippserie wurden 200 Kopien jedes Referenzvolumens erzeugt. Die Partikel in den Volumen wurden mit einem zuvor definierten anisotrop verteilten Rotationssatz gedreht und zufällig translatiert. Der Satz an Rotationen simuliert eine bevorzugte Orientierung der Partikel in der Eisschicht der Probe.

3. Insgesamt wurden 20 Kippserien mit je 400 Subtomogrammen erzeugt. Innerhalb jeder simulierten Kippserie wurde zu jedem der 400 Subvolumen ein zufällig erzeugtes Volumen mit Gaußschem Rauschen addiert. Die Amplitude des Rauschens war so skaliert, dass innerhalb der Kippserie ein definiertes Signal-Rausch-Verhältnis entstand. In 20 Abstufungen wurden so Kippserien mit einem Signal-Rausch-Verhältnis von 0.0025 bis 1.0 erzeugt.

4. Für alle 8.000 Subvolumen wurde eine tomographische Kippserie von -60° bis +60° simuliert, indem die rotierten Phantom Strukturen mit einem passenden *Missing Wedge* Volumen gefaltet wurden.

Die Datensätze wurden mit vier verschiedenen Klassifikationsverfahren analysiert:

1. PCA kombiniert mit *K-Means-Clustering* (Dabei wurde die Implementierung aus dem EM-Package verwendet [Hegerl, 1996]. Bei diesem Verfahren wurde in der Metrik keine Korrektur für den *Missing Wedge* verwendet. Alle Partikel wurden im initialen Schritt (vor der Klassifikation) auf eine einzige Referenz aus weißen Gaußschem Rauschen aligniert.)

2. Maximum CCF-basierte Multi-Referenz-Klassifikation ohne Korrektur für den *Missing Wedge*

3. Maximum CCF-basierte Multi-Referenz-Klassifikation mit der *Compound Wedge Metric*

4. MLTOMO

Für jede Methode und jedes Signal-Rausch-Verhältnis wurde der gemittelte Klassifikationsfehler und der Standardfehler über fünf unabhängige Klassifikationsläufe ermittelt. Insgesamt wurden 20 Signal-Rausch-Verhältnisse, je fünf Läufe für vier verschiedene Verfahren durchgeführt. Damit gingen in Summe 400 Klassifikationen mit je 400 Partikeln in den Vergleich ein. Der Klassifikationsfehler wurde wie folgt ermittelt:

$$\text{Klassifikationsfehler} = \frac{\text{Anzahl falsch klassifizierter Partikel}}{\text{Gesamtanzahl an Partikeln}} \; 100 \qquad (5.1)$$

Jeder Lauf startete mit zwei initialen Referenzen (mit Ausnahme der ersten Methode) bestehend aus unabhängigem zufälligen weißen Gaußschem Rauschen (Abb.: 5.3).

Alle vier Algorithmen berechnen das richtige Klassifikationsergebnis bei einem Signal-Rausch-Verhältnis von 1.0. Der Klassifikationsfehler liegt bei 0. Bei einem Signal-Rausch-Verhältnis von 0.25, ordnet der PCA-basierte Ansatz 10 % aller Subtomogramme in die falsche Partikel Klasse ein (Abb.: 5.3-1). Bei einem Signal-Rausch-Verhältnis von 0.1 werden 10 % aller Subtomogramme von der Maximum CCF-basierten Multi-Referenz-Klassifikation ohne *Missing Wedge* Korrektur falsch klassifiziert (Abb.: 5.3-2). Bei der Kippserie mit einem Signal-Rausch-Verhältnis von 0.01, werden 180 Partikel (von isgesamt 400 Partikeln) von der PCA basierten Klassifikation und der Maximum CCF-basierten Multi-Referenz-Klassifikation falsch eingeordnet, was einem Klassifikationsfehler von 45 % entspricht (Abb.: 5.3-1,2,3). Bei diesem Signal-Rausch-Verhältnis zeigte der vorgestellte MLTOMO Algorithmus einen Klassifikationsfehler kleiner 5 % ((Abb.: 5.3-4). Bis zu einem

Signal-Rausch-Verhältnis von 0.007 erzielte MLTOMO akzeptable Ergebnisse mit einem Klassifikationsfehler kleiner 25 %. Bis zu einem Signal-Rausch-Verhältnis von 0.003 zeigte MLTOMO immer noch bessere Ergebnisse im Vergleich zu den anderen Ansätzen.

5.2 Analyse experimenteller Daten

Bei der Analyse der experimentellen Daten werden die Ergebnisse der Untersuchung von drei verschiedenen Komplexen beschrieben. Bei den ersten beiden Strukturen, dem 26S Proteasom und dem Ribosome wurde MLTOMO für die Mittelung der Strukturen eingesetzt, um zu zeigen, wie robust der ML-Algorithmus ist, wenn keine Informationen über initiale Referenzen vorliegen. In der Analyse der dritten Struktur, dem Thermosom, identifiziert MLTOMO durch Klassifikation eine Konformationsänderung der Komplexe. Für die vorgelagerte Rekonstruktion, Segmentierung und Bearbeitung der Subtomogramme wurde die TOM-Toolbox verwendet [Nickell et al., 2005].

5.2.1 26S Proteasom

In diesem Experiment ist eine aufgereinigte Probe mit isolierten 26S Proteasomen aus der Spezies *Drosophila melanogaster* untersucht worden. Für die Untersuchung mit dem Elektronenmikroskop wurden Proben für die Kippserie ausgewählt, die eine große Anzahl an Partikeln mit intakten doppelten 19S Kappen aufwiesen. Die unter Kryo-Bedingungen aufgenommenen tomographischen Datensätze wurden automatisiert mit einem CM 200 FEG TEM (FEI, Eindhoven, Niederlande) aufgenommen. Die Beschleunigungsspannung betrug 160 kV und die Vergrößerung auf der CCD-Kamera betrug 36.000. Die Vergrößerung entspricht einer Objekt-Pixelgröße von 0.39 nm. Der Defokus lag zwischen -2.5 und -3.0 nm. Der Kippwinkelbereich betrug -60° bis +60° mit einem Inkrement von 5° [Nickell et al., 2007]

Dreidimensionale Rekonstruktionen wurden mittels gewichteter Rückprojektion berechnet. In einer Übersichtsrekonstruktion wurden Positionen von Partikeln manuell ausgewählt und Subtomogramme in der Größe 128x128x128 Voxel in höhere Auflösung rekonstruiert. Für die Untersuchung mit MLTOMO wurden 153 ausgewählte Subtomogramme für die Analyse herangezogen, die auf eine Größe von

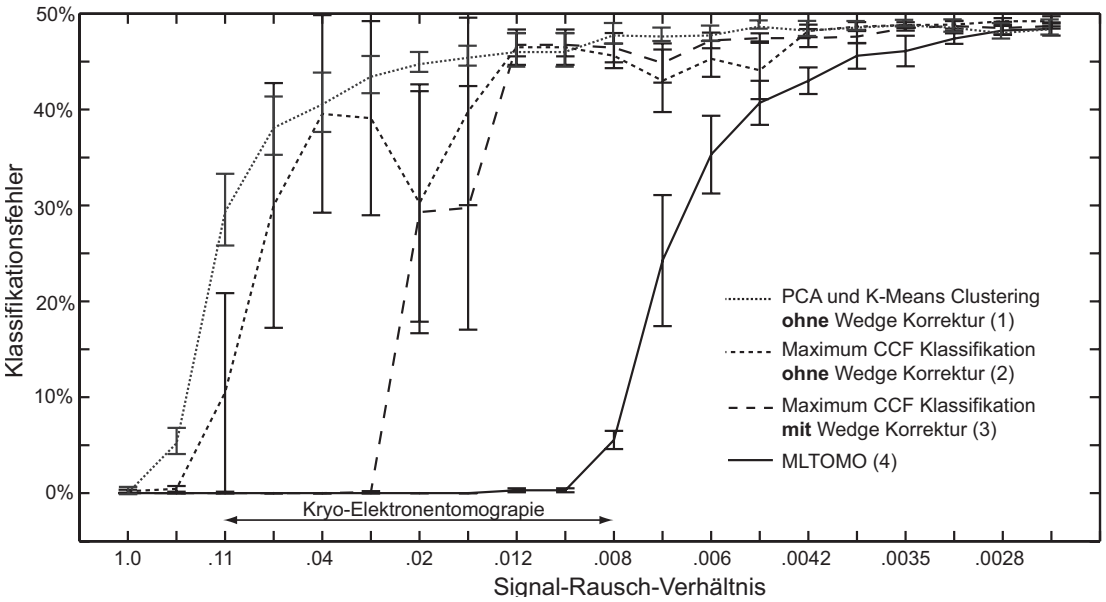


Abbildung 5.3: Auswertung des Klassifikationsfehlers verschiedener Verfahren bei unterschiedlichen Signal-Rausch-Verhältnissen. Ein Datensatz von 400 Subtomogrammen mit zwei unterschiedlichen Partikeln wurde mit Signal-Rausch-Verhältnissen von 0.0025 bis 1.0 simuliert. Der simulierte Kippwinkel reichte von -60° bis +60°. Das Signal-Rausch-Verhältnis in Kryo-Elektronentomogrammen liegt ungefähr zwischen 0.01 und 0.1. Es ist die Klassifikationsgenauigkeit von vier verschiedenen Algorithmen durch die Messung des Klassifikationsfehlers (Gl.: 5.1) der verschiedenen Verfahren dargestellt. Der Fehlerbalken zeigt die Standardabweichung aufgrund unterschiedlicher unabhängiger Messungen mit dem gleichen Verfahren und dem gleichen Signal-Rausch-Verhältnis. (1) PCA kombiniert mit *K-Means Clustering* ohne *Missing Wedge* Korrektur (gepunktete Linie), (2) Maximum CCF-basierte Multi-Referenz-Klassifikation ohne (eng gestrichelte Linie) und (3) mit *Missing wedge* Korrektur (weit gestrichelte Linie), und (4) MLTOMO (durchgezogene Linie). Gaußsches weißes Rauschen wurde als initiale Referenz verwendet. Bei einem Signal-Rausch-Verhältnis von 1.0 waren alle Klassenzuordnungen aller Algorithmen richtig - was einem Klassifikationsfehler von 0 % entspricht (ein Fehler von 50 % bedeutet in diesem Test zufälllige Klassifikation). Bei einem Signal-Rausch-Verhältnis von 0.01 zeigten der PCA-basierte Ansatz und beide Maximum CCF-basierten Multi-Referenz-Klassifikationen einen Fehler von 45 % (1,2,3). Der Klassifikationsfehler des MLTOMO Algorithmus lag hier bei einem Wert kleiner 5 %. Bis zu einem Signal-Rausch-Verhältnis von 0.07 berechnet MLTOMO akzeptable Ergebnisse (4).

64x64x64 Voxel 'gebinnt' (gefiltert) wurden. Die Alignierung wurde mit einem Volumen von Gaußschem weißen Rauschen gestartet. Nach den ersten vier Iterationen hat MLTOMO bereits eine zylinderförmige Struktur gemittelt. Nach der 17. Iteration ist die Struktur des 26S Proteasoms deutlich zu erkennen (Abb.: 5.4). Der Algorithmus hat ohne weitere Kenntnis über die Struktur des 26S Proteasom, die richtige Alignierung der Subtomogramme gefunden.

5.2.2 Ribosomen in Zellen von *Spiroplasma citri*

Für diesen Versuch wurden Subtomogramme aus Tomogrammen von ganzen Zellen von *Spiroplasma citri* verwendet. Die Kippserien für die tomographische Rekonstruktion wurden mit einem FEI Tecnai F30 Polara TEM mit 300 kV aufgenommen. Das Mikroskop war mit einer 2k Gatan CCD Kamera mit Energiefilter ausgestattet. Die Vergrößerung wurde auf 27.500 eingestellt und erzeugte damit eine Objekt-Pixelgröße von 0.46 nm in der Aufnahme. Die Tomogramme wurden unter Kryo-Bedingungen mit einem Defokus zwischen -4 nm and -8 mm aufgenommen. Der Kippwinkelbereich lief von -60° bis +60°. Mehrere Tomogramme wurden unter verschiedenen Bedingungen aufgenommen (Abb.: 5.5). Die Ribosomen wurden mit MOLMATCH [Frangakis et al., 2002] automatisch selektiert [Fleischer, 2010]. Es wurden 1.121 ausgeschnittene Subtomogramme untersucht. Da die Subtomogramme aus einem zellulären Tomogramm entnommen wurden, war das Signal-Rausch-Verhältnis niedriger als bei einer gereinigten Probe. Aus diesem Grund war eine größere Anzahl an Subtomogrammen/Partikeln als Grundlage für die Analyse sinnvoll. Als initiale Referenz wurde eine leicht deformierte Kugel verwendet. Bereits nach sieben Iterationen war MLTOMO in der Lage, die Struktur des Ribosoms zu identifizieren (Abb.: 5.6)

5.2.3 Identifizierung von Konformationsänderungen des Thermosom-Komplexes

Die untersuchte Thermosom Probe wurde aus *Thermoplasma acidophilum* Zellen gewonnen (detailliert beschrieben in [Gutsche et al., 2000b]). Aus einem Gramm Zellen wurde in der Regel ein mg Protein extrahiert. Der erste Teil der Probe wurde mit 15 mM Adenosindiphosphat (ADP) inkubiert, der zweite Teil mit 15 mM Adenosintriphosphat (ATP) und der dritte Teil wurde frei von ATP und ADP inkubiert. Die Proben wurden mit Kryo-Elektronenmikroskopie untersucht unter Verwendung eines CM 200 FEG TEM (FEI, Eindhoven, Niederlande) mit 160 kV.

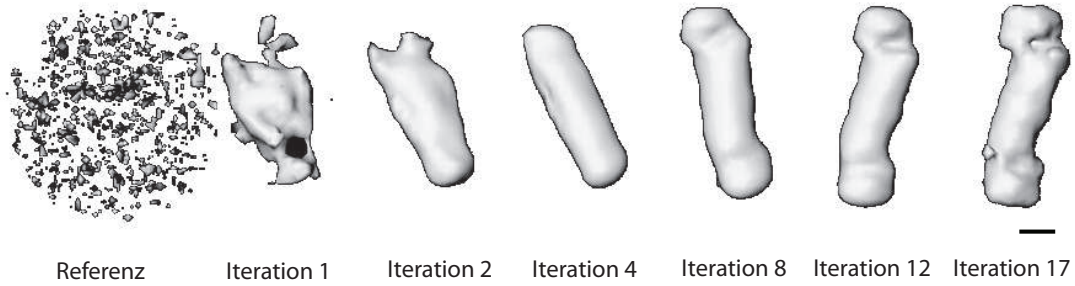


Abbildung 5.4: Alignierung von Subtomogrammen des 26S Proteasom mit MLTOMO (Oberflächendarstellung - Bar entspricht 8 nm).

Die Kippserien wurden unter strikten Nieder-Dosis Bedingungen auf einer 2k CCD Kamera (TVIPS, Gauting, Germany) aufgenommen. Die Objektpixelgröße betrug 0.43 nm. Der Defokus war auf -2.5 nm eingestellt, was einem ersten Nulldurchgang der Kontrasttransferfunktion (CTF) von 2.7 nm entspricht. Der Winkelbereich der Kippserie betrug -60° bis +60°, mit einem Winkelinkrement von 5°.

Für die tomographische Rekonstruktion wurden die Aufnahmen der Kippserie mit Hilfe von Goldmarker aligniert, die der Probe vor der Kryo-Fixierung hinzugegeben wurden. Die Tomogramme wurden über die gewichtete Rückprojektion rekonstruiert. Um individuelle Thermosom-Partikel im Tomogramm zu lokalisieren, wurden Übersichtsrekonstruktionen erstellt und die Position der Partikel wurde visuell und interaktiv lokalisiert. Insgesamt wurden 3.455 einzelne Subtomogramme mit einer Größe von 80x80x80 Voxel rekonstruiert. Um die Laufzeit der Analyse zu reduzieren, wurden alle Subtomogramme auf eine Größe von 32x32x32 reduziert. Entsprechend des beschriebenen experimentellen Aufbaus waren drei Datensätze Gegenstand der Analyse:

1. 1.326 Subtomogramme mit Thermosomn+ADP

2. 1.078 Subtomogramme mit Thermosomn+ATP

3. 1.051 Subtomogramme mit Thermosomn ohne ATP+ADP

Zu Beginn der Analyse wurden die Subtomogramme jedes Datensatzes mit MLTOMO aligniert. Als initiale Referenz wurde ein Volumen mit Gaußschem weißen Rauschen verwendet. Die 3D-Mittelungen zeigten ein typisches Thermosom-Partikel (Abb.: 5.7). Die so ermittelten Strukturen waren die initialen Referenzen für die Klassifikation. In jeder Analyse wurden die entsprechenden Mittelungen verdoppelt

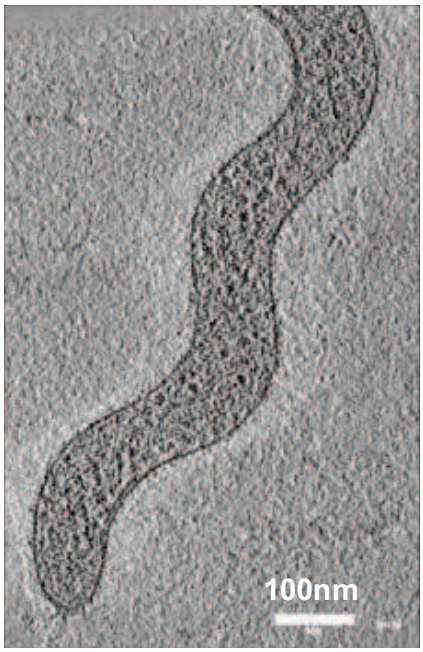


Abbildung 5.5: Schnitt durch eine Übersichtsrekonstruktion von S. citri. Aufnahme wurde unter Kryo-Bedingungen erzeugt.

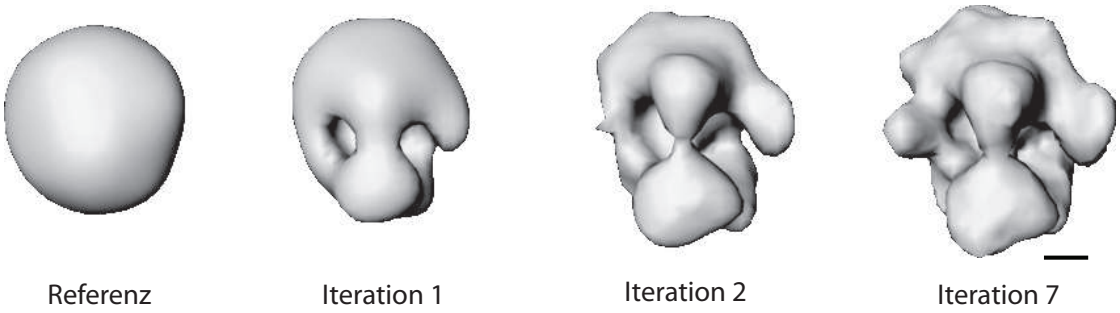


Abbildung 5.6: Alignierung von Subtomogrammen aus *Spiroplasma citri* mit ML-TOMO. Die Alignierung startet mit einer unspezifischen Referenz. Nach der siebten Iteration ist die Struktur des Ribosoms deutlich zu erkennen (Oberflächendarstellung - Bar entspricht 6 nm).

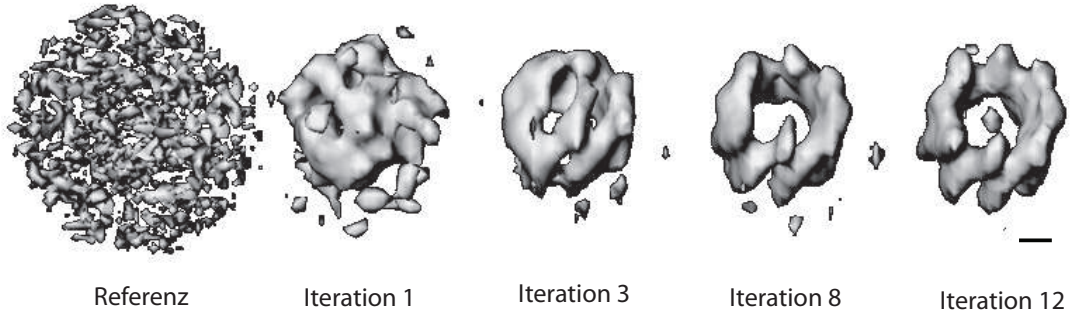


Abbildung 5.7: Alignierung des Thermosom+ADP Datensatzes. Initiale Referenz ist Gaußsches weißes Rauschen. MLTOMO konvergiert nach 12 Iterationen und findet eine typische Thermosom-Struktur (Oberflächendarstellung - Bar entspricht 5 nm).

und zu jedem Strukturvolumen je eins von zwei leicht unterschiedlichen, zufällig erzeugten Volumen aus Gaußschem weißen Rauschen hinzugefügt, um so zwei sich leicht unterscheidende initiale Referenzen zu erhalten. Der Algorithmus iterierte bis der *Likelihood* Wert ein Plateau erreicht hatte und keine signifikante Änderung bei der Alignierung oder Klassifikation mehr festgestellt werden konnte.

MLTOMO war in der Lage, in der Thermosom+ADP Probe zwei unterschiedliche Konformationen zu identifizieren. Die Klassenmittel zeigten eine vierfach symmetrische und eine achtfach symmetrische Struktur (Abb.: 5.9). Die achtfach symmetrische Klasse (Abb.: 5.9 a-c) besteht aus 46 % der Partikel und die vierfach symmetrische Klasse besteht aus anderen 54 % der Partikel (Abb.: 5.9 d-f). Im Vergleich zur Mittelung über alle Partikel verbesserte sich die Ablösung der beiden klassifizierten Mittelungen (Abb.: 5.10). Die Klassifikation der Thermomsome+ATP Daten zeigte eine dominierende achtfach symmetrische Klasse, zu der 90 % der Partikel des Datensatzes zugeordnet wurden (Abb.: 5.8 a, b). Die Probe ohne Zusatz von ATP und ADP zeigte eine dominierende Klasse mit einer klaren vierfachen Symmetrie, in die 90 % der Partikel des Datensatzes fielen (Abb.: 5.8 c, d).

ATP gebundene Thermosom waren bei Raumtemperatur eingefroren. Unter diesen Bedingungen kann lediglich eine Bindung aber keine Hydrolyse stattfinden. Es wurde identifiziert, dass Thermosom+ATP Partikel häufiger in einer achtfachen Symmetrie vorliegen, im Gegensatz zu den Partikeln aus der Probe der keine Nuklotide beigefügt sind. Eine mögliche Erklärung für diese Beobachtung ist, dass die Bindung von ATP die Konformationsänderung in den Untereinheiten bewirkt und sich somit von der Probe ohne Nukliotide unterscheidet (Abb.: 5.8 a-b). Eine

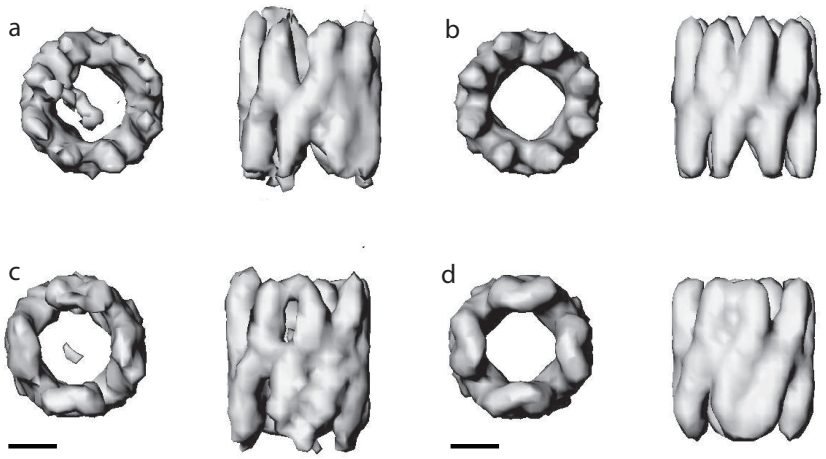


Abbildung 5.8: Thermosom-Konformationen mit und ohne Zugabe von ATP analysiert mit MLTOMO. Oberflächendarstellung (Bar entspricht 5 nm) der dominierenden Klasse der Thermosom+ATP Probe (a, b) und der Probe ohne Zugabe von ATP und ADP (c, d). Gaußsches weißes Rauschen wurde als Referenz für die initiale Alignierung mit MLTOMO verwendet. b und d zeigen die Klassenmittel nach einer Vierfach-Symmetrisierung. Deutlich zu erkennen sind zwei unterschiedliche Konformationen. Die Ergebnisse stimmen mit zuvor durchgeführten Analysen überein [Gutsche et al., 2000b]

partielle Sättigung durch ADP in der Thermosom+ADP Probe könnte der Grund dafür sein, das nur ca. die Hälfte der Partikel in dieser Probe in einer vierfachen Symmetrie vorliegen. Das keine Klasse mit halb oder vollständig geschlossenen Thermosom-Partikeln identifiziert wurde [Clare et al., 2008], liegt wahrscheinlich an der hyperthermophilen Natur von *Thermoplasma acidophilum* Thermosomen [Gutsche et al., 1999,Gutsche et al., 2000a,Bigotti and Clarke, 2005,Bigotti et al., 2006].

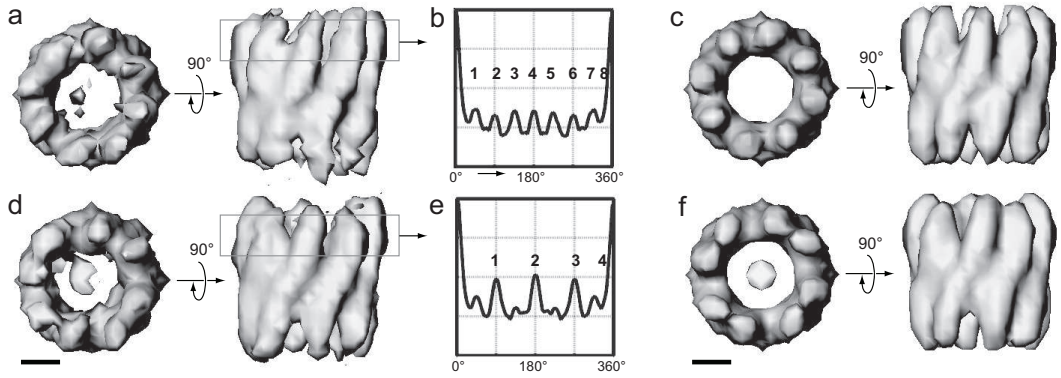


Abbildung 5.9: Identifikation von Thermosom- Konformationsänderungen mit MLTOMO. Oberflächendarstellung (Bar entspricht 5 nm) zweier Klassenmittel (a, d) eines Datensatzes mit 1.326 Thermosom-Subtomogrammen in der Gegenwart von ADP. Gaußsches weißes Rauschen wurde als Referenz für die initiale Alignierung verwendet. Die Symmetrie-Analyse wurde durch eine rotierende Auto-Korrelation im oberen Bereich des Thermosoms ermittelt (a, d). Beide Symmetrie-Plots (b, e) zeigen klare Maxima (acht und vier) und identifizieren damit eine achtfache (b) und eine vierfache (e) Symmetrie. Nach einer vierfach Symmetrisierung für beide Klassenmittel lassen sich zwei verschiedene Konformationen der zwei unterschiedlichen Bindungszustände der Nukleotide des Thermosoms deutlich erkennen, was in genauer Übereinstimmung mit den Ergebnissen vorheriger Analysen steht [Gutsche et al., 2000b].

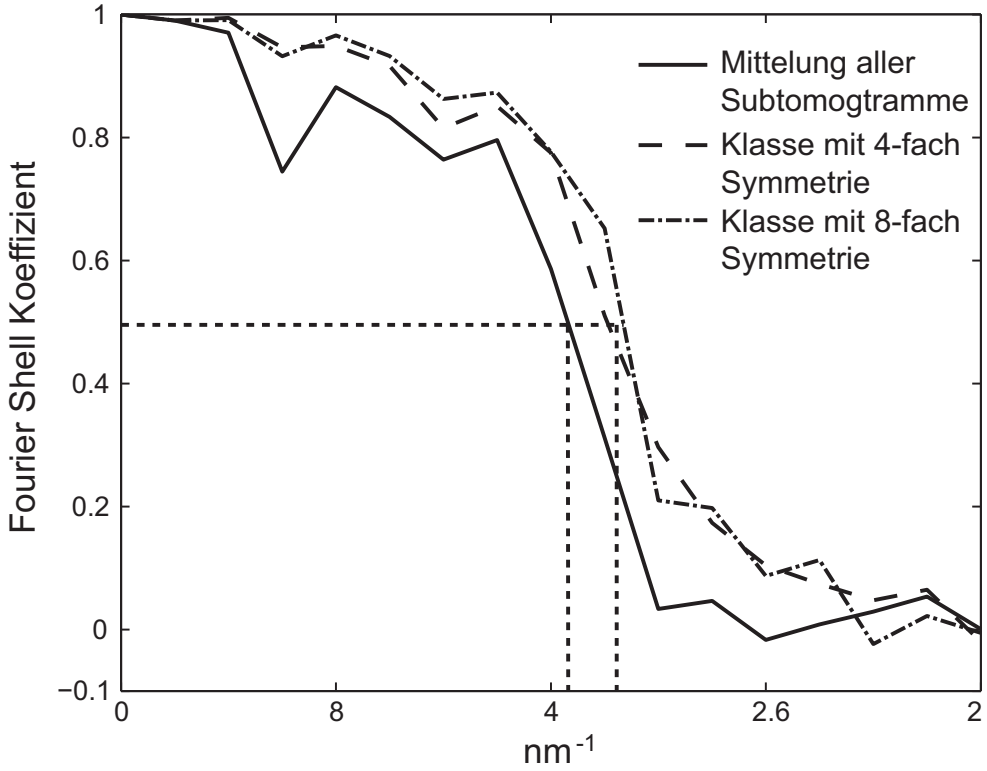


Abbildung 5.10: FSC berechnet aus drei verschiedenen Thermosom+ADP Subtomogramm Mittelungen. Die durchgezogene Linie zeigt die Auflösung der Mittelung über alle Subtomogramme (38.1 Å für das 0.5 Kriterium). Die gestrichelten Linien zeigen die verbesserte Auflösung auf 35 Å nach der Klassifikation der Subtomogramme in zwei Klassen mit unterschiedlichen Konformationen durch MLTOM0.

6 Diskussion und Ausblick

Im Bereich der Einzelpartikel-Elektronenmikroskopie sind Klassifikations- und Mittelungsstrategien mittlerweile eine essentielle Grundlage, um Strukturen molekularer Komplexe zu identifizieren und zu analysieren. Diese Strukturanalyse gibt Aufschluss über die strukturelle Variabilität, die Funktionsweise und der biologischen Aufgabe der untersuchten Komplexe. Um die Abläufe in einer Zelle besser verstehen zu können, ist es wichtig, die Anordnung und die Orientierung der Strukturen in physiologisch definierten Zuständen einer Zelle zu untersuchen. Die Kryo-Elektronentomographie ermöglicht die Aufnahme ganzer Zellen und ihrer Strukturen. Die Entwicklung von erweiterten Analyseverfahren für elektronentomographische Daten ist notwendig, um die die molekulare Interpretation von zellulären Tomogrammen zu verbessern, und damit die biologischen Komplexe im zellulären Kontext zu erforschen.

In dieser Arbeit wurden unterschiedliche Verfahren zur Analyse von Subtomogrammen vorgestellt und diskutiert. Es zeigte sich, dass insbesondere der neu entwickelte 3D Maximum *Likelihood* Algorithmus - MLTOMO - für die Klassifikation und Mittelung von Subtomogrammen gute Ergebnisse lieferte. MLTOMO integriert die Alignierung und die Klassifikation in einem einzigen Prozessschritt der iterativ wiederholt wird. Um die unvollständige Abtastung der Subtomogramme im Fourierraum zu berücksichtigen, werden Informationen über die Geometrie und Orientierungen des *Missing Wedge* gespeichert. Diese '*Missing Wedge* Buchhaltung' erlaubte die Berechnung des *Compound Wedge*, der die Basis für das verwendete Abstandsmass, der *Compound Wedge* Metrik, lieferte. Bei dieser neuen Metrik werden nur die überlappenden Bereiche von Partikel und Referenz für die Vergleichsmessung herangezogen.

Für den Vergleich und die Bewertung unterschiedlicher Analyseverfahren wurden simulierte Phantom-Kippserien mit unterschiedlichen Signal-Rausch-Verhältnissen erzeugt und analysiert. Neben dem Maximum *Likelihood* Algorithmus wurden für den Vergleich die PCA kombiniert mit *K-Means Clustering* und Maximum CCF-basierte Algorithmen verwendet. Dabei wurden verschiedene Eigenschaften der Algorithmen in dedizierten Tests untersucht, wie z.B. der Einfluss der *Compound*

Wedge Metrik auf das Klassifikationsergebnis. Insbesondere die Analyse von Datensätzen mit unterschiedlichen Signal-Rausch-Verhältnissen demonstrierte, dass der ML-Algorithmus präzisere Ergebnisse erreichte als der *PCA*-basierte oder der Maximum CCF-basierte Klassifikationsansatz. MLTOMO arbeitet fehlerfrei bis zu einem Signal-Rausch-Verhältnis von 0.01. Selbst bei einem Signal-Rausch-Verhältnis 0.007 liefert MLTOMO Ergebnisse mit einem akzeptablen Fehler.

Bei der Analyse eines experimentellen tomographischen Datensatzes (Signal-Rausch-Verhältnis von 0.02) bestehend aus Thermosom-Partikeln von *Thermoplasma acidophilum*, konnte MLTOMO zwei unterschiedliche Konformationen makromolekularer Komplexe unterscheiden. Die berechneten 3D Strukturen der Thermosom-Partikel bestätigten vorhergehende Studien, die auf Basis von Bildern aus Einzelpartikelanalysen mit herkömmlichen 2D Mittelungs- und Klassifikationsverfahren berechnet wurden. Bei beiden Analysen wurde eine Konformationsänderung zwischen einer achtfachen und vierfachen Symmetrie der Partikel identifiziert [Gutsche et al., 2001].

MLTOMO ist ein sehr rechenintensives Verfahren. Die Ergebnisse dieser Arbeit wurden überwiegend auf großen Supercomputern mit sehr vielen Prozessoren berechnet. Seit einiger Zeit ist eine rasante Steigerung der Geschwindigkeit für parallelisierte Berechnungen mit Graphik Prozessoren/*Graphics Processing Unit* (GPU) zu beobachten [Bustamam et al., 2011]. GPU-Computer-Cluster sind unter anderem für die schnelle Berechnung der FFT sehr gut geeignet und benötigen bei gleicher Dimensionierung nur einen Bruchteil der Ressourcen von vergleichbaren CPU-Computer-Clustern. Allerdings müssen das Softwaredesign einer Applikation und die Implementierung speziell für die parallele Berechnung auf der GPU entwickelt werden. Software-Pakete für die Alignierung von tomographischen Kippserien auf Basis von GPUs sind bereits verfügbar [Castano-Díez et al., 2010]. Eine Implementierung von MLTOMO für die Berechnung der Klassifikation mit GPUs würde die Anwendung des Verfahrens vereinfachen und neue Dimensionen an Rechenleistung zugänglich machen.

ML-Algorithmen haben das Potential, bereits bestehende Ansätze zur Untersuchung zellulärer Tomographie zu unterstützen bzw. zu erweitern und zu verbessern. So zielt *Visual Proteomics* [Nickell et al., 2006] darauf ab, eine quantitative Beschreibung der Protein Interaktionen zu bestimmen, indem räumliche Koordinaten von molekularen Komplexen in zellulären Tomogrammen identifiziert und lokalisiert werden. Dabei werden die Komplexe in Tomogrammen z.B. über einen Vergleich aller Bereiche, die innerhalb der Zelle liegen, mit einer Referenzstruktur gesucht. Es wird angenommen, dass sich die gesuchten Komplexe in den Bereichen

mit der größten Übereinstimmung befinden. Diese *Template Matching* Strategie erlaubt die Auswahl von spezifischen Subtomogrammen, die dann über nachgelagerte Klassifikation und Mittelung weiter untersucht werden. Die gemittelten Strukturen können dann an die Stelle der zuvor identifizierten Position und Orientierung im Tomogramm gesetzt werden. So entsteht ein synthetisches Tomogramm, das beispielsweise eine Analysen der molekularen Interaktion beziehungsweise der lokalen Häufigkeiten und Konzentrationen der Komplexe erlaubt.

Eine erfolgreiche Analyse der Anordnung und der Orientierung von biologischen Komplexen in der Zelle erfordert jedoch experimentelle Daten, die im Auflösungsbereich der zu detektierenden Strukturen signifikante Unterschiede aufweisen. Mit der derzeitigen mikroskopischen Instrumentation sind sehr dünne Proben ($<$ 500 nm) eine grundlegende Voraussetzung. Aber auch die Analyse dünner Proben wird durch weitere technische Verfahrensgrenzen erschwert. So ist die Gesamtdosis für eine Kippserie intrinsisch begrenzt (ca. 50-100 Elektronen/Å^2) und Aufnahmen unter Niedrigdosisbedingungen sind durch eine sehr niedriges Signal-Rausch-Verhältnis gekennzeichnet. Bedingt durch die Probenpräparation sowie räumliche Ausdehnung der biologischen Strukturen, ist die Objektdicke der Probe an verschiedenen Positionen heterogen. Die geringen Dichteunterschiede zwischen Protein/Makromolekülen und Wasser ergeben nur geringe Kontrastunterschiede in den Aufnahmen. Zusätzlich wird das Signal bei der digitalen Aufzeichnung durch den verwendeten Detektor beinflusst und negativ beeinträchtigt (z.B. Signalverbreiterung, Rauschen etc.). Insgesamt führen diese Faktoren zu einem sehr geringeren und zum Teil variablen Signal-Rausch-Verhältnis in einzelnen elektronenmikroskopischen Aufnahmen und begrenzen die Anwendnung der vorgestellten supramolekularen Analyse zellulärer Tomogramme.

Aufgrund der Qualität der tomographischen Daten können bisherige Analyseverfahren, die z.B. dem *Visual Proteomics* Ansatz zu Grunde liegen, oft nur unzureichende Aussagen liefern. Der häufig verwendete *Template Matching* Ansatz kann momentan nur bekannte Komplexe mit bekannten 3D-Strukturen lokalisieren, unbekannte Komplexe entgehen der Untersuchung. Zusätzlich basieren viele Verfahren auf festen Schwellenwerten. Diese entscheiden beispielsweise, ob es sich an einer Position im Tomogramm um den gesuchten Komplex handelt oder nicht. Durch das geringe Bildsignal und die Varianz des Signal-Rausch-Verhältnisses in den tomographischen Aufnahmen entstehen so zahlreiche falsch-positive Treffer.

Neue technische Lösungen für die Kryo-Elektronentomographie sind in der Entwicklung um bestehende Limitierungen zu umgehen und die Qualität der Tomogramme zu verbessern. Mit einem *Focused Ion Beam* (FIB) lässt sich beispielsweise

die Oberfläche von kryo-fixierten Zellen gezielt abtragen, um die Probendicke für die Elektronentomographie zu optimieren (<500 nm) [Marko et al., 2007]. Das Verfahren erschließt desweiteren die Möglichkeit physiologisch definierte Zustände zu untersuchen, da ausgedehnte homogene Bereiche erzeugt werden können. So lässt sich die Dicke der untersuchten Probenstellen gezielt einstellen [Rigort et al., 2010] und damit die Varianz des Bildsignals aufgrund der Probendicke weitestgehend vermeiden. Des weiteren kommen neuartige Kameras wie der *Falcon(tm) Direct Electron Detector* (FEI, Eindhoven, Niederlande) zum Einsatz, die in der Lage sind, Elektronen direkt (ohne die Umwandlung in Photonen) und ohne merklichen Signalverlust zu detektieren. Diese neuen Detektoren werden bereits verwendet und haben das Potential, das Signal-Rausch-Verhältnis deutlich zu verbessern.

Wahrscheinlichkeitsbasierte Klassifikationsverfahren, wie MLTOMO, eröffnen neue Wege in der Analyse von zellulären Tomogrammen. Im Bereich des Möglichen wären Analysen unabhängig von dem A-priori Wissen über die Proteinstrukturen. Interessante Regionen (bzw. Subtomogramme) könnten über eine strukturlose Such-Referenz (z.B. eine Kugel oder einen Zylinder) im Tomogramm identifiziert oder alternativ interaktiv durch den Anwender ausgewählt werden. Danach erfolgt die Klassifikation der Subtomogramme mit MLTOMO. Der *ML*-Algorithmus ist in der Lage, auch ohne eine definierte initiale Referenz, häufig auftretende Strukturen mit einer hohen Wahrscheinlichkeit zu identifizieren. Anders als bei korrelationsbasierten Ansätzen, gibt es keine festen Schwellenwerte, sondern nur Wahrscheinlichkeiten für die Zugehörigkeit eines Subtomogramms zu verschiedenen Klassen. So können unterschiedliche molekulare Strukturen innerhalb ihrer Klassen kohärent gemittelt und die resultierenden 3D-Klassenmittel über den Vergleich mit bekannten molekularen Komplexen identifiziert werden. Die berechnete Wahrscheinlichkeit für die Klassenzugehörigkeit eines Subtomogrammes, kann als quantitatives Maß für die Qualität der gemittelten Struktur herangezogen werden und lässt weitere Analysen und Interpretationen zu. Zusammen mit den neuesten technischen Entwicklungen in der Kryo-Elektronentomographie kann MLTOMO ein effizientes Analyseverfahren werden, um die Aufklärung biologischer Zusammenhänge in der Zelle weiter voranzutreiben.

Abkürzungsverzeichnis

2D zweidimensional

3D dreidimensional

ADP Adenosindiphosphat

API *Application Programming Interface*

ATP Adenosintriphosphat

CCF Kreuzkorrelationsfunktion

CTF Kontrasttransferfunktion

CCD *Charge-coupled Device*

EM Elektronenmirkroskopie

EM-Algorithmus *Expectation-Maximizaton*-Algorithmus

FIB *Focused Ion Beam*

FFT schnelle Fouriertransformation (*Fast Fourier Transform*)

FSC Fourier-Ring Korrelation

GPU *Graphics Processing Unit*

ML *Maximum Likelihood*

MPI *Message Passing Interface*

PDB *Protein Data Bank*

PCA Hauptkomponentenanalyse

TEM Transmissionselektronenmikroskop

Literaturverzeichnis

[Adrian et al., 1984] Adrian, M., Dubochet, J., Lepault, J., and McDowall, A. W. (1984). Cryo-electron microscopy of viruses. *Nature*, 308(5954):32–36.

[Bartesaghi et al., 2008] Bartesaghi, A., Sprechmann, P., Liu, J., Randall, G., Sapiro, G., and Subramaniam, S. (2008). Classification and 3d averaging with missing wedge correction in biological electron tomography. *J Struct Biol*, 162(3):436–450.

[Bayes, 1991] Bayes, T. (1991). An essay towards solving a problem in the doctrine of chances. 1763. *MD Comput*, 8(3):157–171.

[Best et al., 2007] Best, C., Nickell, S., and Baumeister, W. (2007). *Cellular Electron Microscopy*. Academic Press.

[Bigotti et al., 2006] Bigotti, M. G., Bellamy, S. R. W., and Clarke, A. R. (2006). The asymmetric atpase cycle of the thermosome: elucidation of the binding, hydrolysis and product-release steps. *J Mol Biol*, 362(4):835–843.

[Bigotti and Clarke, 2005] Bigotti, M. G. and Clarke, A. R. (2005). Cooperativity in the thermosome. *J Mol Biol*, 348(1):13–26.

[Bishop, 2006] Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer.

[Bustamam et al., 2011] Bustamam, A., Burrage, K., and Hamilton, N. A. (2011). Fast parallel markov clustering in bioinformatics using massively parallel computing on gpu with cuda and ellpack-r sparse format. *IEEE/ACM Trans Comput Biol Bioinform*.

[Castano-Díez et al., 2010] Castano-Díez, D., Scheffer, M., Al-Amoudi, A., and Frangakis, A. S. (2010). Alignator: a gpu powered software package for robust fiducial-less alignment of cryo tilt-series. *J Struct Biol*, 170(1):117–126.

[Clare et al., 2008] Clare, D. K., Stagg, S., Quispe, J., Farr, G. W., Horwich, A. L., and Saibil, H. R. (2008). Multiple states of a nucleotide-bound group 2 chaperonin. *Structure*, 16(4):528–534.

[Crowther et al., 1970] Crowther, R. A., DeRosier, D. J., and Klug, A. (1970). The reconstruction of a threedimensional structure from projections and its application to electron microscopy. *Proceedings of the Royal Society of London*, 317:319–340.

[Dam et al., 1998] Dam, E., Koch, M., and Lillholm, M. (1998). Quaternions, interpolation and animation. Technical report, DIKU.

[Dempster et al., 1977] Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977). Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society. Series B (Methodological)*, 39, No. 1:1–38.

[DeRosier and Klug, 1969] DeRosier, D. J. and Klug, A. (1969). Reconstruction of three-dimensional structures from electron micrographs. *Nature*, 217:130–134.

[Duda et al., 2001a] Duda, R. O., Hart, P. E., and Stork, D. G. (2001a). *Pattern Classification (2nd Edition)*. Wiley-Interscience.

[Duda et al., 2001b] Duda, R. O., Hart, P. E., and Stork, D. G. (2001b). *Pattern Classification (2nd Edition)*. Wiley-Interscience.

[Dudgeon and Mersereau, 1984] Dudgeon and Mersereau (1984). Multidimensional digital signal processing. *Prentice-Hall, Inc.*, pages 289–311.

[Eppstein, 1992] Eppstein, D. (1992). The farthest point delaunay triangulation minimizes angles. *Comp. Geom. Theory & Applications*, 1:143–148.

[Fleischer, 2010] Fleischer, C. (2010). *Identification and Visualization of Macromolecules in intact cells by Cryo-electron Tomography.* PhD thesis, TU München.

[Frangakis et al., 2002] Frangakis, A. S., Böhm, J., Förster, F., Nickell, S., Nicastro, D., Typke, D., Hegerl, R., and Baumeister, W. (2002). Identification of macromolecular complexes in cryoelectron tomograms of phantom cells. *Proc Natl Acad Sci U S A*, 99(22):14153–14158.

[Frank, 2002] Frank, J. (2002). Single-particle imaging of macromolecules by cryo-electron microscopy. *Annu Rev Biophys Biomol Struct*, 31:303–319.

[Frank et al., 2002] Frank, J., Wagenknecht, T., McEwen, B. F., Marko, M., Hsieh, C.-E., and Mannella, C. A. (2002). Three-dimensional imaging of biological complexity. *J Struct Biol*, 138(1-2):85–91.

[Frigo and Johnson, 2005] Frigo, M. and Johnson, S. G. (2005). The design and implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231. Special issue on "Program Generation, Optimization, and Platform Adaptation".

[Förster, 2005] Förster, F. (2005). *Quantitative Analyse von Makromolekülen in Kryoelektronentomogrammen mittels Korrelationsmethoden.* PhD thesis, Technischen Universität München.

[Förster et al., 2008] Förster, F., Pruggnaller, S., Seybert, A., and Frangakis, A. S. (2008). Classification of cryo-electron sub-tomograms using constrained correlation. *J Struct Biol*, 161(3):276–286.

[Gutsche et al., 1999] Gutsche, I., Essen, L. O., and Baumeister, W. (1999). Group ii chaperonins: new tric(k)s and turns of a protein folding machine. *J Mol Biol*, 293(2):295–312.

[Gutsche et al., 2001] Gutsche, I., Holzinger, J., Rauh, N., Baumeister, W., and May, R. P. (2001). Atp-induced structural change of the thermosome is temperature-dependent. *J Struct Biol*, 135(2):139–146.

[Gutsche et al., 2000a] Gutsche, I., Holzinger, J., Rössle, M., Heumann, H., Baumeister, W., and May, R. P. (2000a). Conformational rearrangements of an archaeal chaperonin upon atpase cycling. *Curr Biol*, 10(7):405–408.

[Gutsche et al., 2000b] Gutsche, I., Mihalache, O., Hegerl, R., Typke, D., and Baumeister, W. (2000b). Atpase cycle controls the conformation of an archaeal chaperonin as visualized by cryo-electron microscopy. *FEBS Lett*, 477(3):278–282.

[Haller, 2008] Haller, T. (2008). Multireference alignment of 3d data from electron tomography. Master's thesis, TU München.

[Hamilton, 1844] Hamilton, W. R. (1844). On quaternions, or on a new system of imaginaries in algebra. *Philosophical Magazine*, 25, n3:489–495.

[Harauz and van Heel, 1986] Harauz, G. and van Heel, M. (1986). Exact filter for general geometry three-dimensional reconstruction. *Optik*, 73:146–156.

[Hart, 1968] Hart, R. (1968). Electron microscopy of unstained biological material: the polytropic montage. *Science*, 159:1464–1467.

[Hegerl, 1996] Hegerl (1996). The em program package: A platform for image processing in biological electron microscopy. *J Struct Biol*, 116(1):30–34.

[Hegerl and Hoppe, 1976] Hegerl and Hoppe, W. (1976). Influence of electron noise on three-dimensional image reconstruction. *Z. Naturforschung*, 31a:1717–1721.

[Hoppe and Hegerl, 1980] Hoppe, W. and Hegerl, R. (1980). Three-dimensional structure determination by electron microscopy (nonperiodic specimens). *Springer-Verlag, Berlin Heidelberg New York*, Bd. 13 d. Reihe Topics in Current Physics Computer Processing of Electron Microscope Images.

[James W. Cooley, 1965] James W. Cooley, J. W. T. (1965). An algorithm for the machine calculation of complex fourier series. *Math. Comput.*, 19:297–301.

[Karney, 2007] Karney, C. F. (2007). Quaternions in molecular modeling. *Journal of Molecular Graphics and Modelling*, 25(5):595–604.

[Kuipers, 2002] Kuipers, J. B. (2002). *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace and Virtual Reality*. Princeton University Press.

[Lucic et al., 2005] Lucic, V., Förster, F., and Baumeister, W. (2005). Structural studies by electron tomography: from cells to molecules. *Annu Rev Biochem*, 74:833–865.

[Marko et al., 2007] Marko, M., Hsieh, C., Schalek, R., Frank, J., and Mannella, C. (2007). Focused-ion-beam thinning of frozen-hydrated biological specimens for cryo-electron microscopy. *Nat Methods*, 4(3):215–217.

[McEwen et al., 1995] McEwen, B. F., Downing, K. H., and Glaeser, R. M. (1995). The relevance of dose-fractionation in tomography of radiation-sensitive specimens. *Ultramicroscopy*, 60(3):357–373.

[McIntosh, 2001] McIntosh, J. R. (2001). Electron microscopy of cells: a new beginning for a new century. *J Cell Biol*, 153(6):F25–F32.

[Nickell, 2001] Nickell, S. (2001). *Elektronentomographische Abbildung eiseingebetteter prokaryotischer Zellen*. PhD thesis, Technische Universität München.

[Nickell et al., 2005] Nickell, S., Förster, F., Linaroudis, A., Net, W. D., Beck, F., Hegerl, R., Baumeister, W., and Plitzko, J. M. (2005). Tom software toolbox: acquisition and analysis for electron tomography. *J Struct Biol*, 149(3):227–234.

[Nickell et al., 2006] Nickell, S., Kofler, C., Leis, A. P., and Baumeister, W. (2006). A visual approach to proteomics. *Nat Rev Mol Cell Biol*, 7(3):225–230.

[Nickell et al., 2007] Nickell, S., Mihalache, O., Beck, F., Hegerl, R., Korinek, A., and Baumeister, W. (2007). Structural analysis of the 26s proteasome by cryo-electron tomography. *Biochem Biophys Res Commun*, 353(1):115–120.

[Nitsch et al., 1998] Nitsch, M., Walz, J., Typke, D., Klumpp, M., Essen, L. O., and Baumeister, W. (1998). Group ii chaperonin in an open conformation examined by electron tomography. *Nat Struct Biol*, 5(10):855–857.

[Palade, 1952] Palade, G. E. (1952). A study of fixation for electron microscopy. *J Exp Med*, 95(3):285–298.

[Penczek et al., 1995] Penczek, Pawel, Marko, M., Buttle, K., and Frank, J. (1995). Double-tilt electron tomography. *Ultramicroscopy*, 60:393–410.

[Penczek et al., 1992] Penczek, P., Radermacher, M., and Frank, J. (1992). Three-dimensional reconstruction of single particles embedded in ice. *Ultramicroscopy*, 40(1):33–53.

[Plaisier et al., 2007] Plaisier, J. R., Jiang, L., and Abrahams, J. P. (2007). Cyclops: new modular software suite for cryo-em. *J Struct Biol*, 157(1):19–27.

[Radon, 1917] Radon, J. (1917). Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. *Berichte Sächsische Akadamie der Wissenschaften, Leipzig, Math.-Phys. Kl.*, 69:262–277.

[Richardson et al., 1960] Richardson, K. C., Jarett, L., and Finke, E. H. (1960). Embedding in epoxy resins for ultrathin sectioning in electron microscopy. *Stain*

Technol, 35:313–323.

[Rigort et al., 2010] Rigort, A., Bäuerlein, F. J. B., Leis, A., Gruska, M., Hoffmann, C., Laugks, T., Böhm, U., Eibauer, M., Gnaegi, H., Baumeister, W., and Plitzko, J. M. (2010). Micromachining tools and correlative approaches for cellular cryo-electron tomography. *J Struct Biol*, 172(2):169–179.

[Roseman, 2000] Roseman, A. M. (2000). Docking structures of domains into maps from cryo-electron microscopy using local correlation. *Acta Crystallogr D Biol Crystallogr*, 56(Pt 10):1332–1340.

[Saxberg and Saxton, 1981] Saxberg, B. and Saxton, W. (1981). Quantum noise in 2d projections and 3d reconstructions. *Ultramicroscopy*, 6:85–90.

[Scheres et al., 2009] Scheres, S. H. W., Melero, R., Valle, M., and Carazo, J.-M. (2009). Averaging of electron subtomograms and random conical tilt reconstructions through likelihood optimization. *Structure*, 17(12):1563–1572.

[Scheres et al., 2005] Scheres, S. H. W., Valle, M., Nu?ez, R., Sorzano, C. O. S., Marabini, R., Herman, G. T., and Carazo, J.-M. (2005). Maximum-likelihood multi-reference refinement for electron microscopy images. *J Mol Biol*, 348(1):139–149.

[Schölkopf and Smola, 2002] Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels, Support Vector Machines, Regularization, Optimiztion, and Beyond*. The MIT Press.

[Schmid and Booth, 2008] Schmid, M. F. and Booth, C. R. (2008). Methods for aligning and for averaging 3d volumes with missing data. *J Struct Biol*, 161(3):243–248.

[Sigworth, 1998] Sigworth, F. J. (1998). A maximum-likelihood approach to single-particle image refinement. *J Struct Biol*, 122(3):328–339.

[Sipser, 1997] Sipser, M. (1997). *Introduction to the Theory of Computation*, chapter 7.1 Measuring complexity, pages 226–228. PWS Publishing.

[Stölken, 2008] Stölken, M. (2008). Entwicklung eines maximum-likelihood algorithmus zur klassifikation von partikeln in kryo-elektronenmikroskopischen tomogrammen. Master's thesis, Zentrum für Bioinformatik Universität Hamburg.

[Vicci, 2001] Vicci, L. (2001). Quaternions and rotations in 3-space: The algebra and its geometric interpretation. Technical report, Department of Computer Science, UNC at Chapel Hill.

[Walz et al., 1997] Walz, Typke, Nitsch, Koster, Hegerl, and Baumeister (1997). Electron tomography of single ice-embedded macromolecules: Three-dimensional alignment and classification. *J Struct Biol*, 120(3):387–395.

[Winkler et al., 2009] Winkler, H., Zhu, P., Liu, J., Ye, F., Roux, K. H., and Taylor, K. A. (2009). Tomographic subvolume alignment and subvolume classification applied to myosin v and siv envelope spikes. *J Struct Biol*, 165(2):64–77.

Danksagung

Diese Doktorarbeit wurde von April 2008 bis November 2011 in der Abteilung Molekulare Strukturbiologie des Max-Planck-Instituts für Biochemie, Martinsried, durchgeführt. Ich möchte mich bei allen Kollegen für das gute Arbeitsklima bedanken.

Herrn Prof. Dr. W. Baumeister danke ich für die Möglichkeit, diese Arbeit in seiner Abteilung zu erstellen.

Stephan Nickell und Jürgen Plitzko danke ich für die Betreuung und Unterstützung meiner Arbeit.

Florian Beck danke ich für seine motivierende Unterstützung und guten Ideen bei allen Fragen zur Elektronenmikroskopie und zur Bildverarbeitung.

Thomas Haller danke ich für seine Hilfe bei der Implementierung von MLTOMO und für anregende Diskussionen zu allen mathematischen Themenstellungen.

Reiner Hegerl danke ich für sein Interesse, seine Antwortbereitschaft und kritisches Korrekturlesen der Arbeit.

Sjors Scherres danke ich für anregende Diskussionen zur Maximum-Likelihood Methode.

Birgit Book und Inga Wolf danke ich für die Unterstüzung bei allen administrativen Belangen.

Heidi Severin und Isabel Börgen danke ich für die grammatikalische Korrektur der Arbeit.

Agnes Scherling und Jenny Stölken danke ich für die Organisation rund um meine beiden Kinder, die mir die Zeit verschafft hat, diese Arbeit fertig zu stellen.

Meinen Eltern danke ich für die großzügige Unterstützung während der gesamten Zeit.

Meinem Bruder danke ich für die Motivation, mit der ich auch in schwierigen Zeiten geschafft habe, weiter zu machen.

Meinen beiden Töchtern Ada Oranna und Milla Margarethe danke ich für die Geduld, die sie mit mir hatten und dafür, dass sie da sind und mir Freude bereiten.