Untersuchungen zur (An-) Ordnung von Entwurfsentscheidungen

Rauscher, Reinhard; Univ. Hamburg; Hamburg

Zusammenfassung:

In vielen Entwürfen wird die Frage nach den zu fällenden Entwurfsentscheidungen oft intuitiv behandelt. Auch die sich bietenden Alternativen werden bereits bei der Auswahl stark durch mehr oder weniger unbewußt einfließende Erfahrungen aus z.T. völlig anderen Entwürfen geprägt. Im Paper werden anhand praktischer Aufgabenstellungen Entwurfsentscheidungen, deren Alternativen und deren gegenseitige Abhängigkeiten behandelt. Darauf aufbauend wird ein System zur 'Anordnung von Entwurfsentscheidungen' vorgestellt.

1. Motivation/Einleitung

Der Entwurf hochintegrierter Systeme war ursprünglich ein sehr zeitintensiver Prozeß, der nur von wenigen hochspezialisierten Entwerfern, die über detaillierte Kenntnisse zu den zugrundeliegenenden Technologien verfügten, durchgeführt werden konnte. Mittlerweile hat sich der Entwurf - nicht zuletzt aufgrund sich durchsetzender Modelle (z. B. Standardzell-Entwurf [6], Meet-in-the-Middle [7]) - mehr und mehr zu einer meist intuitiv vorgenommenen oder aber aufgrund vorheriger Entwürfe als optimal angesehener Abfolge von Entwurfsentscheidungen entwickelt. Dementsprechend definieren Leung und Shanblatt ([4]):

> "... viewing design as a decision making process. That is, system designers must structure the design process so that the solution space is manageable and design alternatives are consciously sought and evaluated."

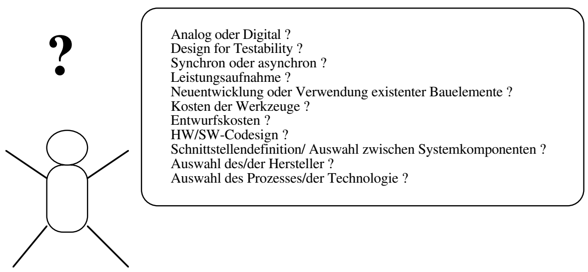


Abbildung 1: Konfusion des Entwerfers

Bild 1 zeigt die - ungeordneten - Fragestellungen, die auf einen potentiellen Entwerfer quasi einstürmen. Nach einer vom Entwerfer vorzunehmenden Anordnung hat er für jede der Entwurfsentscheidungen eine der sich bietenden Alternativen auszuwählen. Rabaey ([7]) sagt hierzu:

> "enormous dimensions of the design space have to be explored."

Eine a priori Festlegung der Abfolge der Entwurfsentscheidungen wäre wünschenswert und dringend erforderlich, wird es i.a. nicht geben. Knapp und Parker ([5]):

> "...and also the ordering of the decisions can be permutated as well. This makes an a priori ordering of design decisions of doubtful utility."

In der Literatur wird die Priorisierung einzelner Entwurfsentscheidungen teilweise völlig konträr diskutiert ([8, 3]). Sogar die Anzahl der zu treffenden Entwurfsentscheidungen differiert von wenigen (10) bis zu 1000. Knapp und Parker ([5]):

> "In a design of practical size, there are ten, hundreds or thousands of such decisions. Deciding which decision to make next and predicting the outcome of each individual decisions is a combinatoric problem."

Zusammenfassend kann festgestellt werden, daß sowohl Anzahl als auch Art und Abfolge der Entwurfsentscheidungen zweifelsfrei an die jeweiligen Aufgabenstellung angepaßt werden muß. Verdeutlicht wird diese Aussage speziell durch den Vergleich der Aufgabenstellungen, die aus der Mikrosystemtechnik entstehen mit solchen aus der mittlerweile klassischen Mikroelektronik. Während in letzterem Fall die Entwurfsmethodik/Entwurfsentscheidungen anhand mittlerweile allgemein anerkannter Modelle (siehe z.B [2, 1, 7]) relativ festgelegt ist, herrscht in der Mikrosystemtechnik noch weitgehend Unklarheit.
Betrachtet man nochmals die in Bild 1 dargestellten Entwurfsentscheidungen, so können sich einige am jeweiligen Entwurf orientierte Restriktionen für das Fällen von Entwurfsentscheidungen ergeben.
Dies sind entweder der Zwang zum PARALLELEN Festlegen von Entwurfsentscheidungen oder aber der Zwang zum SERIELLEN Festlegen.
Beispiele für das parallele Festlegen könnten im o.g. Bild die Aspekte *synchron oder asynchron* und *Leistungsaufnahme* sein. Dies könnte u.a. genau dann der Fall sein, wenn über verschiedene Arten der Spannungsversorgung noch nicht entschieden ist.
Ein weiteres Beispiel wäre *Analog oder Digital* und *Schnittstellendefinition/-auswahl zwischen Systemkomponenten.* Genau dieser Fall ist parallel zu entscheiden, falls sowohl geeignete digitale als auch geeignete analoge Schnittstellen zur Auswahl stehen.
Beispiele für das serielle Festlegen könnten in o.g. Bild sein:
Neuentwicklung oder Verwendung existenter Bauelemente,
Auswahl des/der Hersteller, Auswahl des Prozesses/der Technologie
Zunächst wird die erstgenannte Entscheidung zu treffen sein und nur, falls die Entscheidung zugunsten der Neuentwicklung geht, ist die Auswahl des Herstellers, der Tools und damit des Prozesses bzw. der Technologie zu treffen.
Im folgenden wird ein Konzept und ein daraus abgeleitetes Tool vorgestellt, das zur strukturierten Abfolge von Entwurfsentscheidungen führen soll. Anforderungen an ein derartiges Tool sind (speziell in Hinblick auf das oben gesagte):

- Inkrementelle Erweiterbarkeit um zusätzliche Entwurfsentscheidungen
- Inkrementelle Erweiterbarkeit um zusätzliche Abhängigkeiten (parallel oder seriell)

Ziel ist es, eine Anordnung aller Entwurfsentscheidungen zu treffen, so daß alle spezifizierten Anforderungen an Serialität und Parallelität erfüllt werden.
Falls aber nicht alle Anforderungen erfüllt werden können - die Möglichkeit hierfür resultiert bereits aus dem zweiten Item der vorherigen Liste -, gilt es, Verstöße gegen einzelne Anforderungen zu **gewichten.**
Aufgabe ist dann, die Anordnung zu finden, bei der die Summe der Gewichte der verletzten Anforderungen minimal ist.

2. Mathematisches Modell

In diesem Kapitel soll ein mathematisches Modell für die Abfolge von zu treffenden Entwurfsentscheidungen entwickelt werden.
Hierzu soll durchgängig ein Beispiel verwendet werden.

Beispiel 1:
Gegeben sei folgende Menge von Entwurfsentscheidungen: {E1, E2, E3, E4 und E5}. Diese Menge entstammt übrigens - als Teilmenge - aus einem real durchgeführten Entwurf. Zur Verdeutlichung kann hier eine eineindeutige Abbildung der e_i auf Entwurfsentscheidungen vorgenommen werden:

- E1 entspricht "Design for testability".
- E2 entspricht "Synchroner or asynchroner Entwurf?".
- E3 entspricht "Leistungsaufnahme (Power consumption)".
- E4 entspricht "Kosten der einzusetzenden Entwurfswerkzeuge".
- E5 entspricht "Auswahl des Herstellers und/oder des Produktionsprozesses".

Es gelten folgende Anforderungen an die Abfolge:
E1 vor E2 mit Gewicht 1.
E3 vor E5 mit Gewicht 10.
E2 vor E5 mit Gewicht 4.
E2 parallel mit E4 mit Gewicht 3.
E2 parallel mit E3 mit Gewicht 6.
E4 parallel mit E5 mit Gewicht 2.

Def. 1: Sei G=(V,E) ein gerichteter, gewichteter Graph. Jeder Knoten $v \in V$ entspricht genau einer der zu treffenden Entwurfsentscheidungen. G heißt *Reihenfolgegraph (S)*, falls für jede Kante $e = (x, y) \in E \subset VxV$ mit dem Gewicht $g(e) \in \mathbb{N}$ gilt:
- x ist vor y auszuführen.
- Das Gewicht der sequentiellen Ausführung von x und y entspricht dem Wert g(x,y).

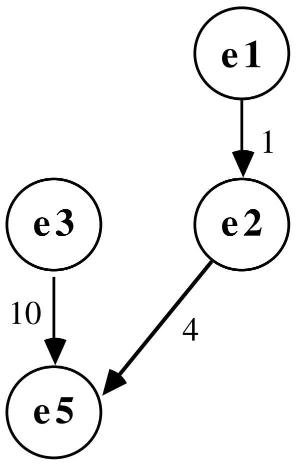


Abbildung 2: Der Reihenfolgegraph zu dem Beispiel 1

Def. 2: Sei $G = (V, E)$ ein gerichteter, gewichteter Graph. Jeder Knoten $v \in V$ entspricht genau einer der zu treffenden Entwurfsentscheidungen.
G heißt *Parallelitätsgraph (P)*, falls für jede Kante $e = (x, y) \in E \subset VxV$ mit dem Gewicht $p(e) \in \mathbb{N}$ gilt:
- x ist gleichzeitig mit y auszuführen.
- Das Gewicht der parallelen Ausführung von x und y entspricht dem Wert p(x,y).

Bem.: Der Parallelitätsgraph könnte (aufgrund der Symmetrieeigenschaft) auch als ungerichteter Graph definiert werden. Daß hier vorerst noch die Asymmetrie berücksichtigt werden muß, liegt daran, daß Entwerfer u.U. unsymmetrische Parallelitäten spezifizieren könnten.

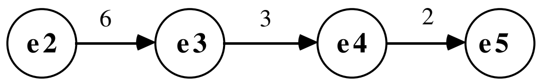
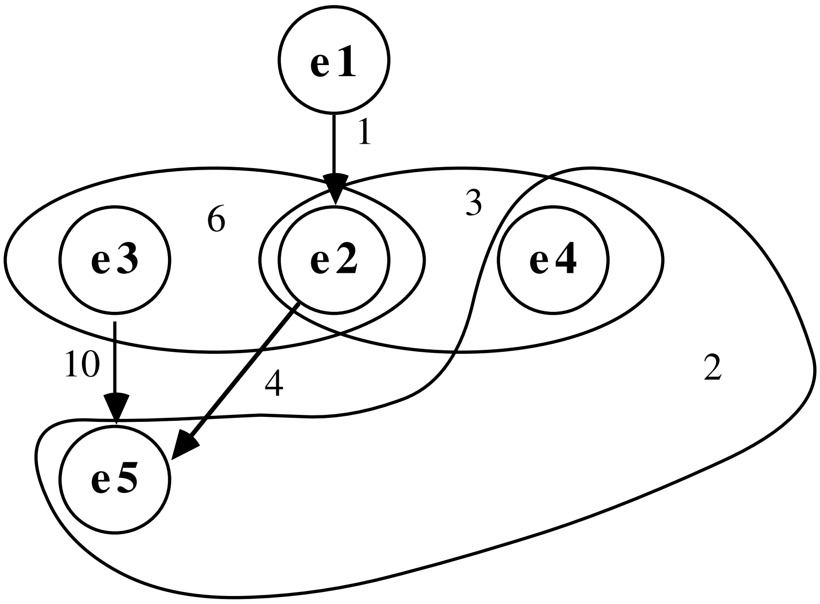


Abbildung 3: Der Parallelitätsgraph zu dem Beispiel 1

Beispielsweise könnte eine Spezifikation wie folgende vorliegen:
"E1 parallel mit E2 mit Gewicht $g1$" und
"E2 parallel mit E1 mit Gewicht $g2$" $(g1 \neq g2)$

Def. 3: Seien je ein Reihenfolgegraph $(R) = (V, E')$ und Parallelitätsgraph $(P) = (V, E'')$ zu einer Menge V von Entwurfsentscheidungen gegeben. Hieraus ergibt sich der *Abhängigkeitsgraph (S) = (V,ES, EP)* wie folgt:
$\forall (x, y) \in V \text{x} V$ gilt:
$(x, y) \in ES \Leftrightarrow (x, y) \in E'$
$(x, y) \in EP \Leftrightarrow (x, y) \in E''$.

Ferner werden die Kantengewichte von den Graphen R und P übernommen. Sofern eine Unterscheidung zwischen Kantengewichten in ES und denen in EP erforderlich ist, wird im folgenden $g_{ES}(e)$ und $g_{EP}(e)$ verwandt.



Legende: Parallelitätskanten werden durch Ovale markiert,
Serialitätskanten durch normale Striche.

Abbildung 4: Der Abhängigkeitsgraph zu dem Beispiel 1

Durch die Forderung nach inkrementeller Erweiterbarkeit bzgl. der Anforderungen an die Parallelität bzw. Serialität von Entwurfsentscheidungen ergibt sich zwangläufig, daß mit Inkonsistenzen bei dem Abhängigkeitsgraphen gerechnet werden muß. Ein Beispiel hierfür ist in Bild 4 zu sehen. Abhilfe könnte hier die transitive Hülle schaffen.

Def. 4: Sei $G = (V, E)$ ein gerichteter, ungewichteter Graph. Die *transitive Hülle T=(V,E')* wird wie folgt definiert:
$\forall (x, y) \in E \Rightarrow (x, y) \in E'$
$\forall (x, y) \in E'$ und $(y, z) \in E' \Rightarrow (x, z) \in E'$.

Bem.: Bei ungewichteten Graphen ist die Definition der transitiven Hülle eindeutig und sinnvoll (sowohl für Parallelitätsgraphen als auch für Serialitätsgraphen).

Bem.: Bei gewichteten Gaphen ist die Def. der transitiven Hülle nicht eindeutig. Es gibt gute Gründe für z.B. folgende Ansätze:

- Das neue Gewicht zu (x,z) ergibt sich durch das Minimum der Gewichte zu (x,y) und (y,z). Allgemeiner:

  Sei $w = (x_1, ..., x_k)$ ein Weg von x_1 nach x_k mit $(x_i, x_{i+1}) \in E$, $\forall i \in \{1, ..., k-1\}$.

  Das neue Gewicht zu $(x_1, x_k) \in E'$ (im folgenden auch als Gewicht des Weges w referenziert) ergibt sich durch das Minimum der Gewichte der einzelnen (x_i, x_{i+1}).

- Seien $w_1, ..., w_j$ Wege von x_1 nach x_k. Das neue Gewicht zu $(x_1, x_k) \in E'$ ergibt sich durch das Maximum der Gewichte zu $w_1, ..., w_j$.

Diese Alternativen sind sinnvoll für beide Anforderungsarten (parallel und seriell)!

Def. 5: Sei $S = (V, ES, EP)$ gegeben. Dann können die Funktionen
output,input,omi: $V \rightarrow \mathbb{Z}$
$output(v) := \sum_{(v,y) \in ES} g_{ES}((v, y))$
$input(v) := \sum_{(x,v) \in ES} g_{ES}((x, v))$
$omi(v) := output(v) - input(v)$

Die Definition von *omi* (kurz für 'output minus input') kann wie folgt auf Knotenmenge erweitert werden: *setomi:* $P(V) \rightarrow \mathbb{Z}$:
$setomi(T) := \sum_{v \in T} omi(v), T \subset V$
setomiint: $P(V)xP(V) \rightarrow \mathbb{Z}$ wird wie folgt definiert: Seien $V1, V2 \subset V, V1 \cap V2 = \phi$.

$$setomiint(V1, V2) := \sum_{(v,y) \in ES \wedge v \in V2, y \in V1} g(v, y) - \sum_{(v,y) \in ES \wedge v \in V1, y \in V2} g(v, y)$$

Informal beschreibt *setomiint* die Summe der gewichteten Kanten zwischen den Mengen $V1$ und $V2$. Im Beispiel wäre dementsprechend:

output(E2) = 4
input(E2) = 1
omi(E2) = 3
setomi(E2, E3)= 14 -1 = 13
setomiint: ({ E2, E3, E4}, { E5}) = -14 + 0 = -14

Bei der Parallelität ergibt sich durch parallel spezifizierte Entwurfsentscheidungen die Problemstellung, daß u.U. als parallel spezifizierte oder aber durch Bildung der transitiven Hülle der Parallelitäten entstandene parallel auszuführende Entwurfsentscheidungen wiederum zu sequentialisieren sind.
In Bild 5 wird hierzu ein Beispiel vorgestellt.
In Bild 5 ist die Sequentialisierung der Menge {E2, E3, E4, E5} in {E2, E3, E4} und (sequentiell) {E5} aufgeführt. Falls nun eine Partition $P1, P2$ der Menge $\{E1, ..., Ek\}$ existiert mit:
$|setomi(P1) - setomi(P2)| > |setomiint(P1, P2)|$, dann kann eine Sequentialisierung in erst P1, dann P2 u.U. hilfreich sein, zumindest aber dem Anwender vorgeschlagen werden.

Def. 6: Zielfunktion. Gegeben sei ein Abhängigkeitsgraph $(S) = (V, ES, EP)$.
Gesucht ist eine lineare Quasiordnung ($\gg$) der Knotenmenge V, so daß

1. Ei $\gg$ Ej und Ej $\gg$ Ei $\Leftrightarrow (Ei, Ej) \in EP$.
2. $\sum_{(x,y) \in ES \text{ und } (x \not\gg y)} g(x, y)$ ist minimal.

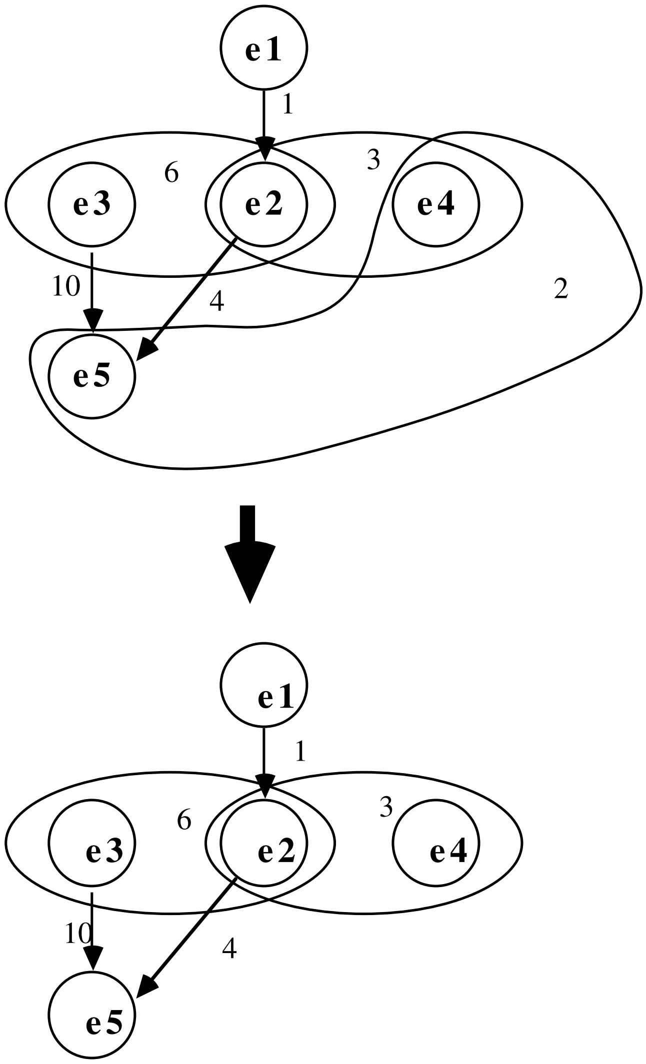


Abbildung 5: Beispiel einer Sequentialisierung parallel spezifizierter Entscheidungen

Def. 7: Meilensteine. Gegeben sei ein Abhängigkeitsgraph $(S) = (V, ES, EP)$ und eine lineare Quasiordnung $(\gg)$ der Knotenmenge V. Ein Knoten $v \in V$ heißt *Meilenstein*, falls gilt:
$\nexists v1, v2 \in V \mathrm{x} V$: $\quad v1 \gg v \gg v2$ *und* $(v1, v2) \in ES$ oder $(v2, v1) \in ES$.

3. Implementierung

Basierend auf oben geschilderten mathematischem Modell wurde eine Implementierung vorgenommen. Eingabe ist zum einen eine sehr einfache und auch einfach inkrementell erweiterbare Beschreibung der Anforderungen an serielle /parallele Festlegung der Entwurfsentscheidungen. Diese Eingabe erfolgt über eine ASCII-Datei. Ferner ist interaktiv durch den Benutzer festzulegen, ob und welche der Alternativen für die Bildung der transitiven Hülle ausgewählt werden. Danach wird interaktiv nach dem Wunsch einer Sequentialisierung von parallel spezifizierten Entwurfsentscheidungen gefragt. Eine weitere Sequentialiserung wird nicht vorgenommen. Dementsprechend wird im Folgenden mit PD jeweils eine Menge von parallel definierten Entwurfsentscheidungen ('parallel decisions') bezeichnet.
Im Falle, daß zur aktuellen Entwurfsentscheidung keine weitere als parallel spezifiziert ist, gilt $|PD| = 1$.

Die Heuristik zur Ermittlung der geeigneten Abfolge von *PDs* ist folgende:
1. Menge der noch anzuordnenden PD: $M := V$
2. repeat
3. Ermittlung von $setomi(A) \subset M$ für alle PD A
4. B:= die PD $\subset M$, für die gilt: setomi(B) ist maximal;
5. Die Menge B wird als nächste festzulegende Entscheidungsmenge ausgegeben.
6. $M := M \setminus B$
until $M = \phi$
7. STOP

Diese Heuristik ist in einem zweiten Schritt dahingehend verfeinert worden, daß die Menge M dreigeteilt wurde:

1. Menge der PDs, die keine Vorgänger besitzen.
2. Menge der PDs, die sowohl Vorgänger als auch Nachfolger besitzen.
3. Menge der PDs, die nur Nachfolger besitzen.

Danach werden gemäß der o.g. Heuristik zunächst ausschließlich die Entwurfsentscheidungen aus 1. betrachtet und sukzessive angeordnet, danach die aus der Menge 2 und zum Schluß die aus der 3. Menge. Ausgabe ist u.a. die partiell geordnete Liste der Entwurfsentscheidungen.

4. Ergebnisse anhand einiger Chip-Entwürfe

Anhand einiger real durchgeführter ASIC-Entwürrfe wurde das System erprobt. Ausgangspunkt war eine Menge von 30 Entwurfsentscheidungen sowie auf die jeweilige Problemstellung abgestimmte Gewichtungen der Abfolge der Entwurfsentscheidungen.
Ergebnisse waren:

1. Tw. wurden nicht alle Entwurfsentscheidungen überhaupt benötigt, d.h. die Entscheidung war für den aktuellen Entwurf entweder bereits durch Vorgaben gefüllt oder völlig irrelevant.
2. Tw. ergaben sich völlig andere Abfolgen als im realen Entwurf. Diese Unterschiede resultierten aus dem "klassischen" Entwurfsvorgehen, das sich sehr stark an dem üblichen Top-Down-Entwurf orientierte.
3. Tw. ergaben sich auch Fragestellungen, die erst durch die Untersuchungen zu den Entwurfsentscheidungen offensichtlich wurden. Diese Fragestellungen wurden im realen Entwurf oft durch eine vom Entwerfer prädestinierte Festlegung a priori festgelegt. Ein Beispiel hierfür ist die Festlegung der Technologie oder des Herstellers.
4. Durch Bildung der transitiven Hülle bzw. Sequentialiserung von parallelen Entscheidungen ergaben sich unterschiedliche Abfolgen von Entwurfsentscheidungen.

Der letzte Punkt soll hier etwas detaillierter behandelt werden. Zur Bewertung des Unterschieds zweier Abfolgen sollen folgende Metriken eingeführt werden:

4.1 Metrik 1

Gegeben seien zwei Anordnungen $A_1 = (p_1, p_2, .., p_k)$ und $A_2 = (p'_1, p'_2, ..., p'_k)$ derselben Menge von PDs mit $|A_1| = |A_2| = k$.
Mit $ord(e, A_i)$ sei die Ordnungsnummer der Pentscheidung e bezüglich der Anordnung A_i bezeichnet. Formal: $ord(e, A_i) = i \Leftrightarrow p_i = e$
Der Abstand zwischen zwei Anordnungen wird definiert wie folgt:

$$D1(A_1, A_2) := \sum_{i:=1}^{k} (|ord(e_i, A_1) - ord(e_i, A_2)|).$$

Nach Metrik 1 stellt sich die Frage nach dem maximalen (**mdist**) und dem durchschnittlichen (**adist**) Abstand.
Der *maximale Abstand* **mdist** kann durch folgende Formel beschrieben werden:

$$mdist := 2 \star \lfloor \frac{k}{2} \rfloor \star \lceil \frac{k}{2} \rceil$$

Dies kann wie folgt bewiesen werden: Man betrachte zwei Permutationen p1, p2 der Zahlen $1, ..., k$:

$$p1 = \begin{pmatrix} 1 & 2 & .. & k-1 & k \\ 1 & 2 & .. & k-1 & k \end{pmatrix} \qquad p2 = \begin{pmatrix} 1 & 2 & .. & k-1 & k \\ k & k-1 & .. & 2 & 1 \end{pmatrix}$$

$mdist = (k-1) + (k-3) + (k-5) .... \quad +(k - 2\lfloor \frac{k}{2} \rfloor + 1) + \quad (k - 2\lfloor \frac{k}{2} \rfloor + 1) + (k - 2\lfloor \frac{k}{2} \rfloor + 3) + ... \quad +(k-3) + (k-1)$

Damit folgt: $\quad mdist = \sum_{i=1}^{\alpha} (k + 1 - 2i) + \sum_{i=\alpha+1}^{k} -(k + 1 - 2i) \qquad$ mit $\alpha = \lfloor \frac{k}{2} \rfloor$
Aus Symmetriegründen folgt :
$mdist = 2 \star \sum_{i=1}^{\alpha} (k + 1 - 2i) = 2 \star \sum_{i=1}^{alpha} (k+1) - 4 \star \sum_{i=1}^{\alpha} i = 2 \star \alpha \star (k+1) - 2 \star \alpha \star (\alpha + 1) = 2 \star \alpha \star (k + 1 - \alpha - 1) = 2 \star \alpha \star (k - \alpha) = 2 \star \lfloor \frac{k}{2} \rfloor \star \lceil \frac{k}{2} \rceil$ q.e.d.

Der *durchschnittliche Abstand* **adist** ergibt sich zu:

$$adist \geq \frac{mdist}{2}.$$

Die Beweisidee hierbei ist, daß zu jeder Permutation $p1$ über k Elementen es genau eine weitere Permutation $p2$ gibt mit $p2 \circ p1 \circ p0 = p3$, wobei

$$p0 = \begin{pmatrix} 1 & 2 & .. & k-1 & k \\ 1 & 2 & .. & k-1 & k \end{pmatrix} \qquad p3 = \begin{pmatrix} 1 & 2 & .. & k-1 & k \\ k & k-1 & .. & 2 & 1 \end{pmatrix}$$

es gilt: $dist(p1, p0) + dist(p2, p0) \geq dist(p3, p0)$.
Damit gilt: $\quad dist(p1, p0) + dist(p2, p0) \geq mdist$.
Damit kann der durchschnittliche Abstand $adist \geq \frac{mdist}{2}$ abgeschätzt werden. q.e.d.

Beispielsweise gilt bei $k = 30$, $adist = 225$.

4.2 Metrik 2

Gegeben seien zwei Anordnungen A_1 und A_2 von PDs mit $k_1 = |A_1| \geq |A_2| = k_2$. Es gelte ferner: $\forall E \in A_1 : \exists E' \in A_2 : E \subset E'$.
Zum Vergleich der beiden Anordnungen nehme man an, daß A_1 und A_2 zu A_1' und A_2' derart verfeinert werden, daß gilt:

1. $|A_1'| = |A_2'|$
2. $\forall\ E \in A_1' : \exists E' \in A_2' : E = E'$.

Danach können die Anordnungen wie mit Metrik 1 behandelt werden:

$$D2(A_1, A_2) := \sum_{i:=1}^{k_1} (|ord(e_i, A_1') - ord(e_i, A_2')|).$$

4.3 Ergebnisse

Zur Evaluierung des vorgestellten Konzepts wurde eine nachträgliche Analyse praktischer Beispiele, die an unserem Institut durchgeführt wurden, vorgenommen.
Die Beispiele sind im einzelnen:

- 'small' (das oben dargestellte, kleine akademische Beispiel)
- 'general' (Ein Satz von allgemeinen Abhängigkeiten zwischen den 30 Entwurfsentscheidungen)
- Vier am FB Informatik, Univ. Hamburg, durchgeführte Entwürfe)

Für jeden dieser Entwürfe wurden folgende Ergebnisse ermittelt:
Für alle vier möglichen Alternativen (mit oder ohne Bildung der transitiven Hülle, mit oder ohne Durchführung einer Sequentialisierung) wurden folgende Werte ermittelt:

1. Die Anordnung (Schedule) der - teilweise parallel vorzunehmenden - Entwurfsentscheidungen.
2. Die Anzahl (eventuell auch die Liste) der Meilensteine
3. Die Anzahl von sequentiell auszuführenden PDs.
4. Der Abstand nach der Metrik $D2$ zwischen je zwei Anordnungen zu derselben Aufgabenstellung.

4.3.1 Ergebnisse zu 'small'

Zu dem kleinen akademischen Beispiel werden zwei Schedules in Abhängigkeit von der Wahl des Entwerfers zu Bildung der transitiven Hülle und der Frage nach einer Sequentialisierung behandelt. Bei Wahl: KEINE transitive Hülle, KEINE Sequentialisierung ergibt sich folgende Anordnung:

- Ausführung von E1.
- Ausführung von E3. Parallel zu E3 auszuführen: E2.
- Ausführung von E4. Parallel zu E4 auszuführen: E5.

Die Anzahl der Meilensteine ist : 2
Die Anzahl der sequentiell auszuführenden PDs ist: 3.

Im folgenden wird eine weitere Schedule vorgestellt, bei der eine andere Festlegung der Parameter (Bildung der transitiven Hülle sowie Durchführung einer Sequentialisierung) gewählt wurde.

Schedule für 'small' (MIT transitiver Hülle, MIT Sequentialisierung):
- Ausführung von E3.
- Ausführung von E1.
- Ausführung von E4. Parallel zu E4 Ausführung von: E2.
- Ausführung von E5.

Für die Ermittlung des Abstandes zwischen zwei Schedules wird folgende Konvention getroffen: Abkürzend für MIT/Ohne transitive Hülle bzw. MIT/OHNE Sequentialisierung wird nur die Belegung beider boolescher Werte angegeben. Beispielsweise bedeutet ny, daß keine transitive Hülle gebildet wurde, aber eine Sequentialisierung vorgenommen wurde.
Die Abstände nach der Metrik $D2$ sind in folgender Tabelle zusammengefaßt:

D2	ny	yn	yy
nn	4	8	4
ny	-	4	0
yn	-	-	8

Jedes '-' in der Tabelle deutet an, daß aufgrund der Symmetrieeigenschaft von $D2$ der Wert nicht berechnet werden mußte. Die obige Tabelle zeigt, daß die Schedules sehr unterschiedlich sind, so daß die Auswirkungen einer - automatisch durchgeführten - Sequentialisierung bzw. Bildung der transitiven Hülle nicht vernachlässigbar sind.

4.3.2 Beispiel: Allgemeine Abhängigkeiten

	nn	ny	yn	yy
Anzahl der sequentiellen PDs	25	26	18	22
Anzahl der Meilensteine	2	2	2	2

D2	ny	yn	yy
nn	44	284	168
ny	-	308	40
yn	-	-	246

4.3.3 Beispiel: Entwurf No. 1 "Skelettierer"

	nn	ny	yn	yy
Anzahl der sequentiellen PDs	22	25	16	18
Anzahl der Meilensteine	3	2	3	3

D2	ny	yn	yy
nn	236	194	186
ny	-	230	224
yn	-	-	12

4.3.4 Beispiel: Entwurf No. 2 "SCHEME-Prozessor"

	nn	ny	yn	yy
Anzahl der sequentiellen PDs	24	26	18	21
Anzahl der Meilensteine	2	2	2	2

D2	ny	yn	yy
nn	54	62	112
ny	-	110	64
yn	-	-	60

4.3.5 Beispiel: Entwurf No. 3 "Krümmungsadaptiver Deltamodulator"

	nn	ny	yn	yy
Anzahl der sequentiellen PDs	27	27	21	22
Anzahl der Meilensteine	2	2	2	2

D2	ny	yn	yy
nn	0	214	128
ny	-	214	128
yn	-	-	228

4.3.6 Beispiel: Entwurf No. 4 "Leistungsmesser"

	nn	ny	yn	yy
Anzahl der sequentiellen PDs	22	22	18	21
Anzahl der Meilensteine	2	2	2	2

D2	ny	yn	yy
nn	186	110	192
ny	-	228	44
yn	-	-	202

5. Weitere Aktivitäten

Als weitere Aktivitäten sind geplant:

- Untersuchungen weiterer Beispiele.
- Untersuchungen zu inkonsistenten Vorgaben, dies wird speziell durch die Forderung nach inkrementeller Erweiterbarkeit erschwert.
- Konzeption, Entwicklung und Bewertung weiterer Heuristiken

Literatur

[1] Erenyi, I.; Pongracz, J.: Quality Control in Textile Industry via Machine Version, Proc. EUROMICRO 91, Wien, Sept. 1991, pp 807-813

[2] Gajski, D.: Silicon Compilation, Addison-Wesley, 1988

[3] Giloi, W.K.; Shriver, B.D.: An Overview of the Computer System Design Process; in: Giloi, W.K. (ed.), Methodologies for Computer System Design, North-Holland, Amsterdam, 1985

[4] Leung, S.; Shanblatt, M.: ASIC System Design with VHDL: A Paradigm, Kluwer Acad. Press., Boston, 1989

[5] Knapp, D.; Parker, A.: The ADAM Design Planning Engine, IEEE Transact. on CAD, Vol. 10, No. 7, July 1991, pp 829-846

[6] Mead, C.; Conway, L.: Introduction to VLSI Systems, Addison-Wesley, Reading, Mass., 1980

[7] Rabaey, J.; De Man, H. et al.: Cathedral-II: A Synthesis System for Multiprocessor DSP Systems, In: Silicon Compilation, D.D. Gajski (ed), Addison-Wesley, 1988, pp 311-360

[8] VLSI DESIGN: Silicon Compilers: Part 2; VLSI DESIGN, Okt. 1987, pp 65-68