

交差点記号化標識の配列ルール
～自動配列プログラムによる検討

首都大学東京 教授 大口 敬
(社)交通工学研究会自主研究代表 松平 健



目的

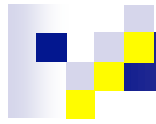
ココ!マークは、現在展開される市内27交差点だけではなく、県内全域、また全国10万の主要交差点全てにまで展開できるように、検討しておく必要がある。

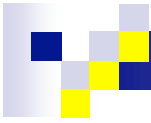
そのため、広域展開時、隣接地域への展開時には、記号化予定の交差点に、自動的に記号を割り振る配列プログラムによって最適な記号配列を行なえるようにする。プログラム内に、配列ルールを組み込むものとする。

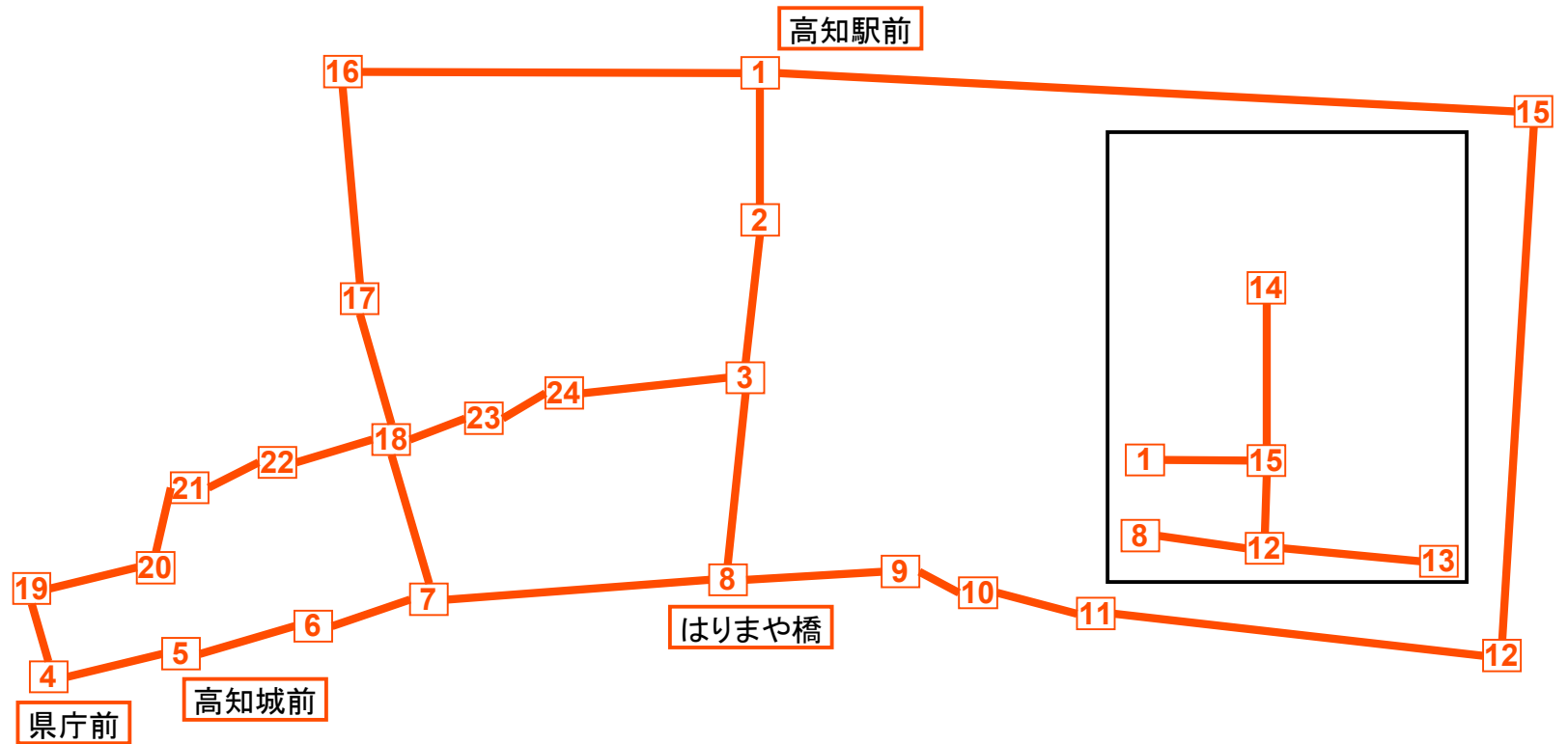
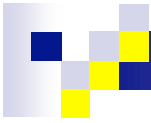


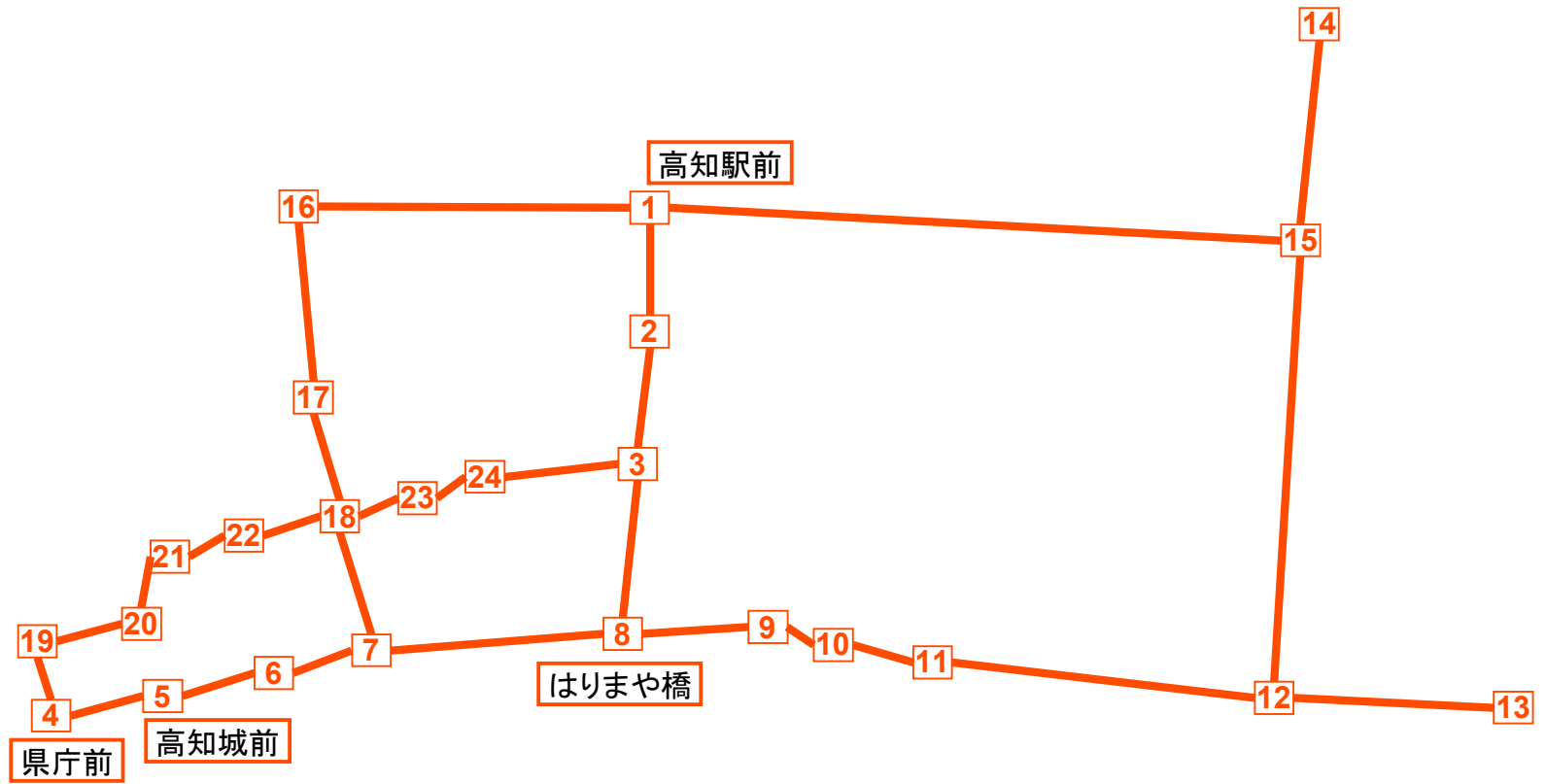
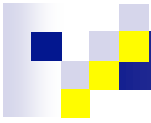
目的

- 記号化標識の自動配列プログラムのプロトタイプを作成する
 - 自動配列ルール：
道路ネットワーク上の対象交差点に、重複も許しながら、できる限り相互に遠くに配列されるように記号を配列する
- 合理的な配列を客観的計算により実現する
 - 同一記号同士が十分に遠くに離れているか
記号化交差点の追加に対して対応可能か。
- 既に設置されている記号化標識の配列を、客観基準により評価・確認する
 - 既に配列したものの妥当性検証
- 他の配列方法との比較検証を行なう
 - 順列配列との比較
 - 重複を許さない配列との比較
- 他地域での配列シミュレーションを行ない、客観基準により評価・確認する
 - 都市中心部等、交差点密度が最も高い地域への配列シミュレーション
 - 分岐数が最も多い交差点周囲への配列シミュレーション
 - 高知50交差点への配列シミュレーション
- 記号化標識の意義に関する再確認
 - 主要地点標識(114-2Aおよび2B)との関係性









配置プログラム

kouchi_24.csv 交差点目印標識配列プログラム

ファイル(F) 配置(A) ヘルプ(H)

ココ！マーク高知(24交差点)

[配置]

	基点	自動	固定	信号名称
1	<input type="checkbox"/>		S	高知駅前
2	<input type="checkbox"/>			
3	<input type="checkbox"/>			
4	<input type="checkbox"/>		K	県庁前
5	<input type="checkbox"/>		J	高知城前
6	<input type="checkbox"/>			
7	<input type="checkbox"/>			
8	<input checked="" type="checkbox"/>		H	はりまや橋
9	<input type="checkbox"/>			
10	<input type="checkbox"/>			
11	<input type="checkbox"/>			
12	<input type="checkbox"/>			
13	<input type="checkbox"/>			

kouchi_24.csv 交差点目印標識配列プログラム

ファイル(F) 配置(A) ヘルプ(H)

ココ！マーク高知(24交差点)

[配置]

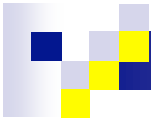
	基点	自動	固定	信号名称
1	<input type="checkbox"/>	S	S	高知駅前
2	<input type="checkbox"/>	E		
3	<input type="checkbox"/>	B		
4	<input type="checkbox"/>	K	K	県庁前
5	<input type="checkbox"/>	J	J	高知城前
6	<input type="checkbox"/>	G		
7	<input type="checkbox"/>	C		
8	<input checked="" type="checkbox"/>	H	H	はりまや橋
9	<input type="checkbox"/>	A		
10	<input type="checkbox"/>	D		

[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
J	2	5 14	8	4.134
K	2	4 13	9	4.061
W	2	12 19	9	2.26

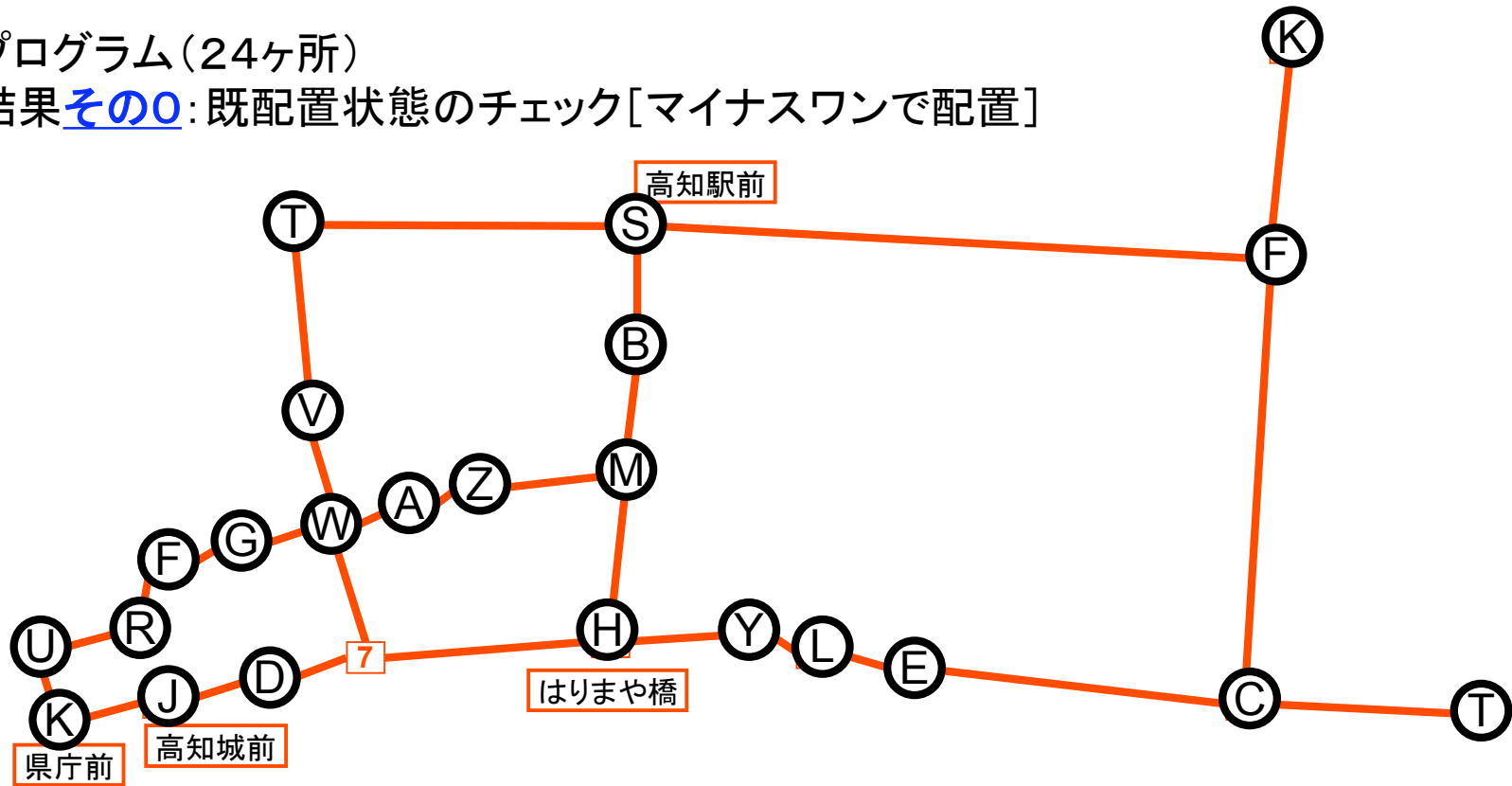
プログラムのルール

1. 基点となる交差点に最初の記号を配置する(基点が指定されていない場合は1番の交差点から)
2. 最も近い交差点に次の未使用記号を配置する
 2. -2 未使用の記号が無い場合はそこから最も遠い(最短経路で通過する交差点数が多い)交差点に配置されている記号を採用する
 2. -3 同じ遠さの交差点が複数ある場合は使用個数の少ない記号を採用する
3. すべての交差点に記号が配置されるまで2. を繰り返す



配置プログラム(24ヶ所)

計算結果その0: 既配置状態のチェック[マイナスイオンで配置]



配置プログラム(24ヶ所)

計算結果その0: 既配置状態のチェック[マイナスワンで配置]

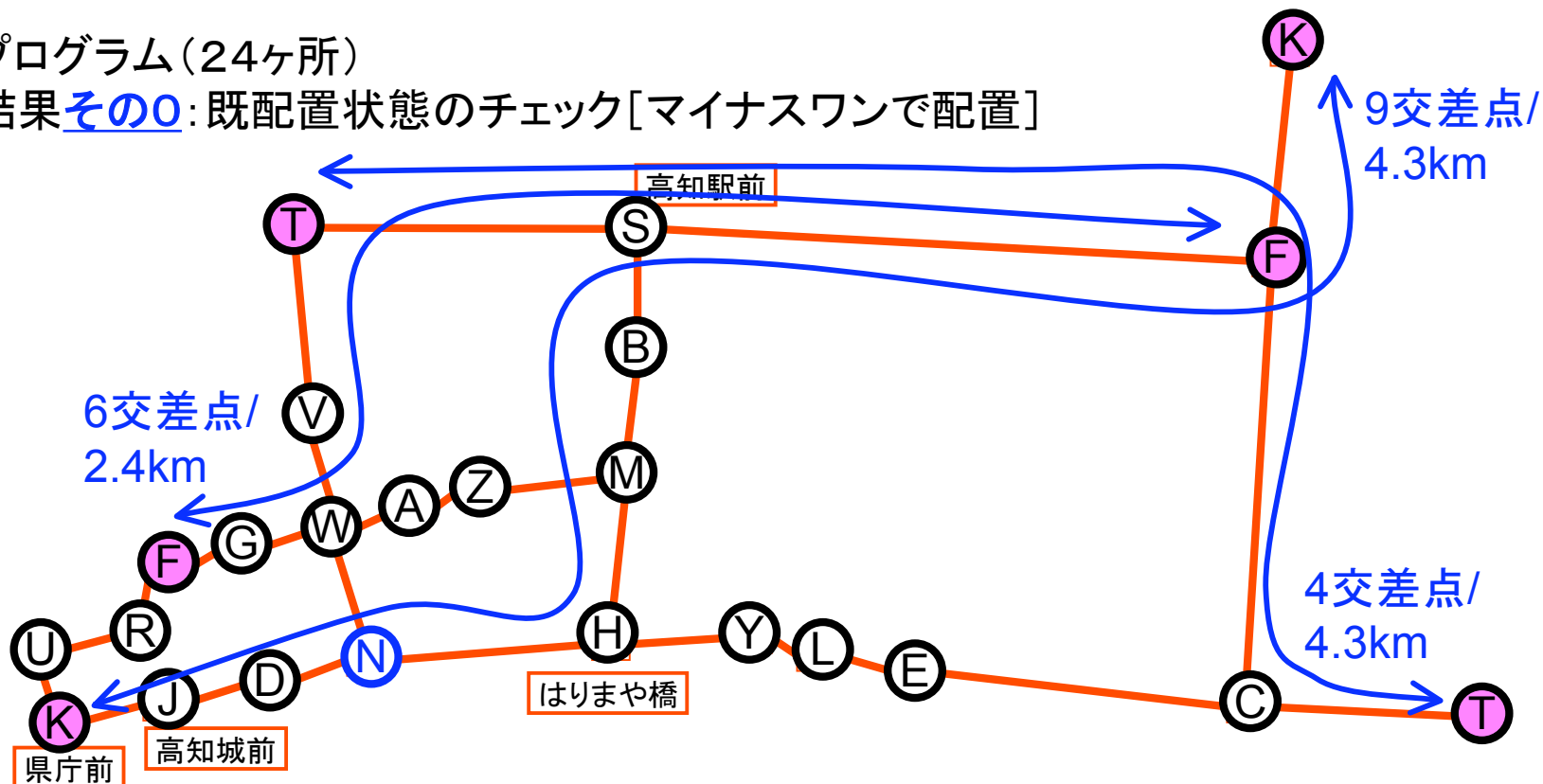
重複状況の計算結果の出力

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
F	2	15 21	6	2.376
K	2	4 14	9	4.323
T	2	13 16	4	4.267

Nが配置されてきた!

配置プログラム(24ヶ所)

計算結果その0: 既配置状態のチェック[マイナスワンで配置]



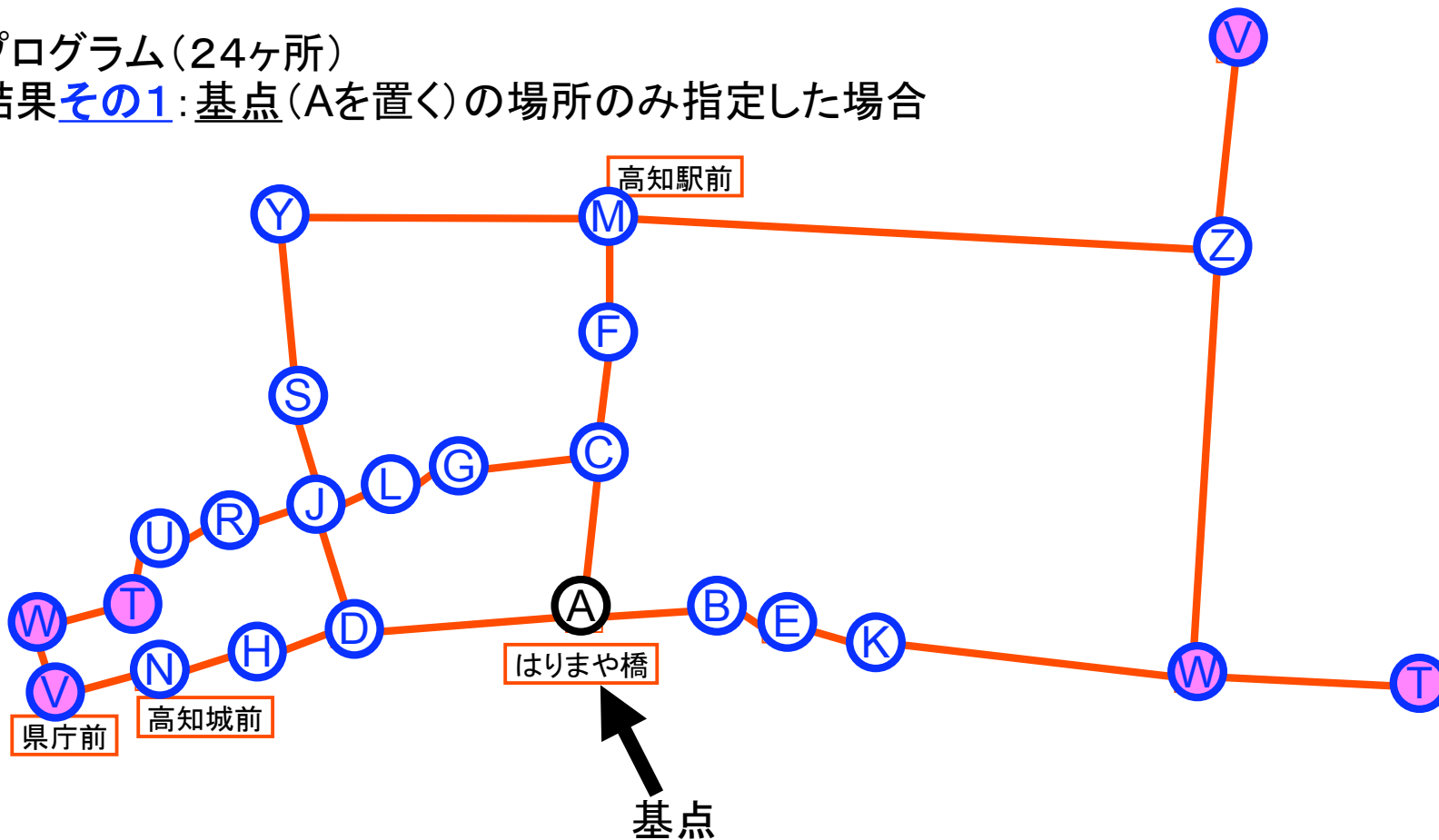
※平均交差点数は、未記号化交差点を含まないもの

※平均交差点数: 6.3
平均距離: 3.65

[重複]				
	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
F	2	15 21	6	2.376
K	2	4 14	9	4.323
T	2	13 16	4	4.267

配置プログラム(24ヶ所)

計算結果その1: 基点(Aを置く)の場所のみ指定した場合



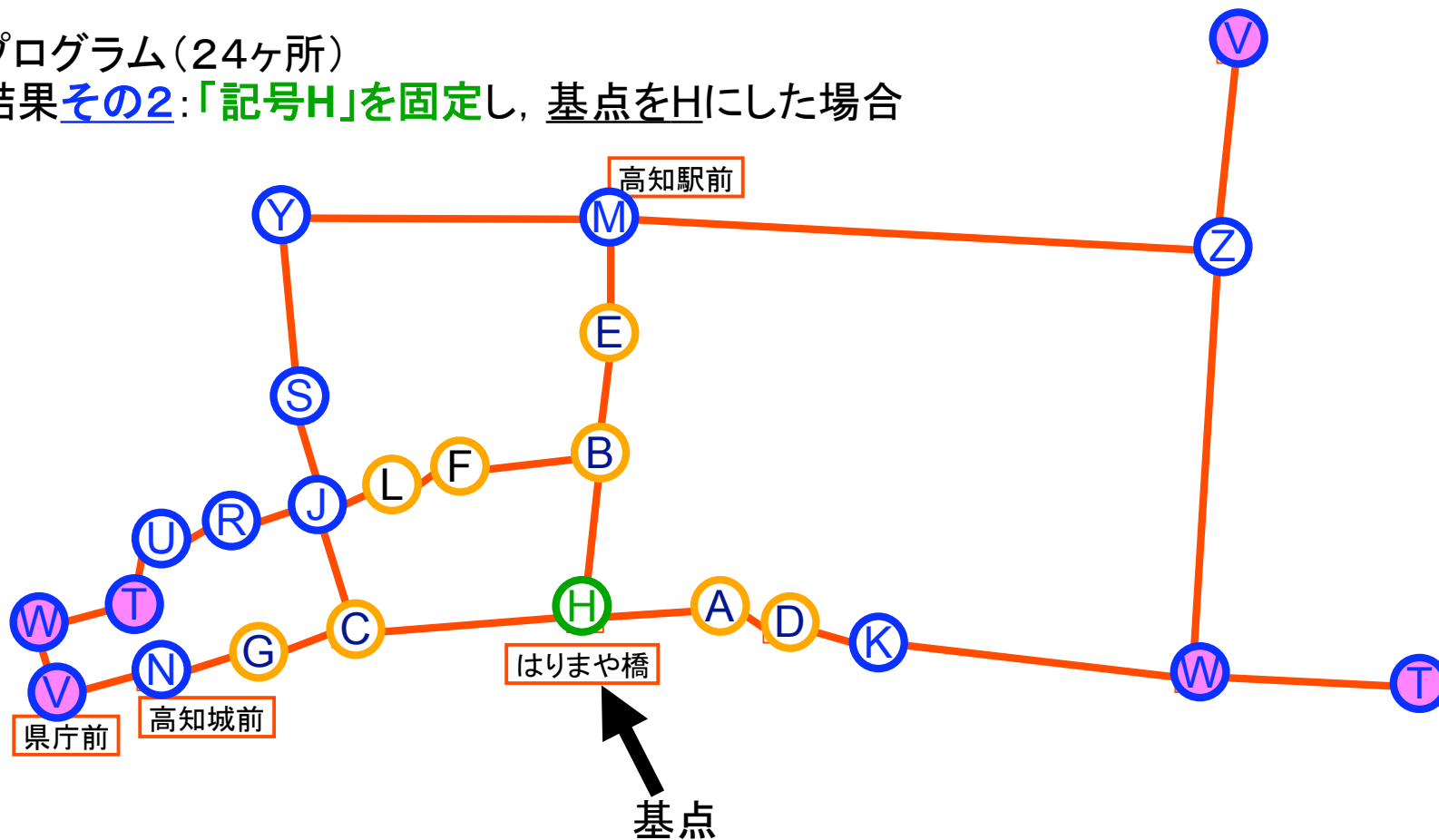
平均交差点数: 9.0
平均距離: 3.53

[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
T	2	13 20	9	3.999
V	2	4 14	9	4.323
W	2	12 19	9	2.26

配置プログラム(24ヶ所)

計算結果その2:「記号H」を固定し、基点をHにした場合



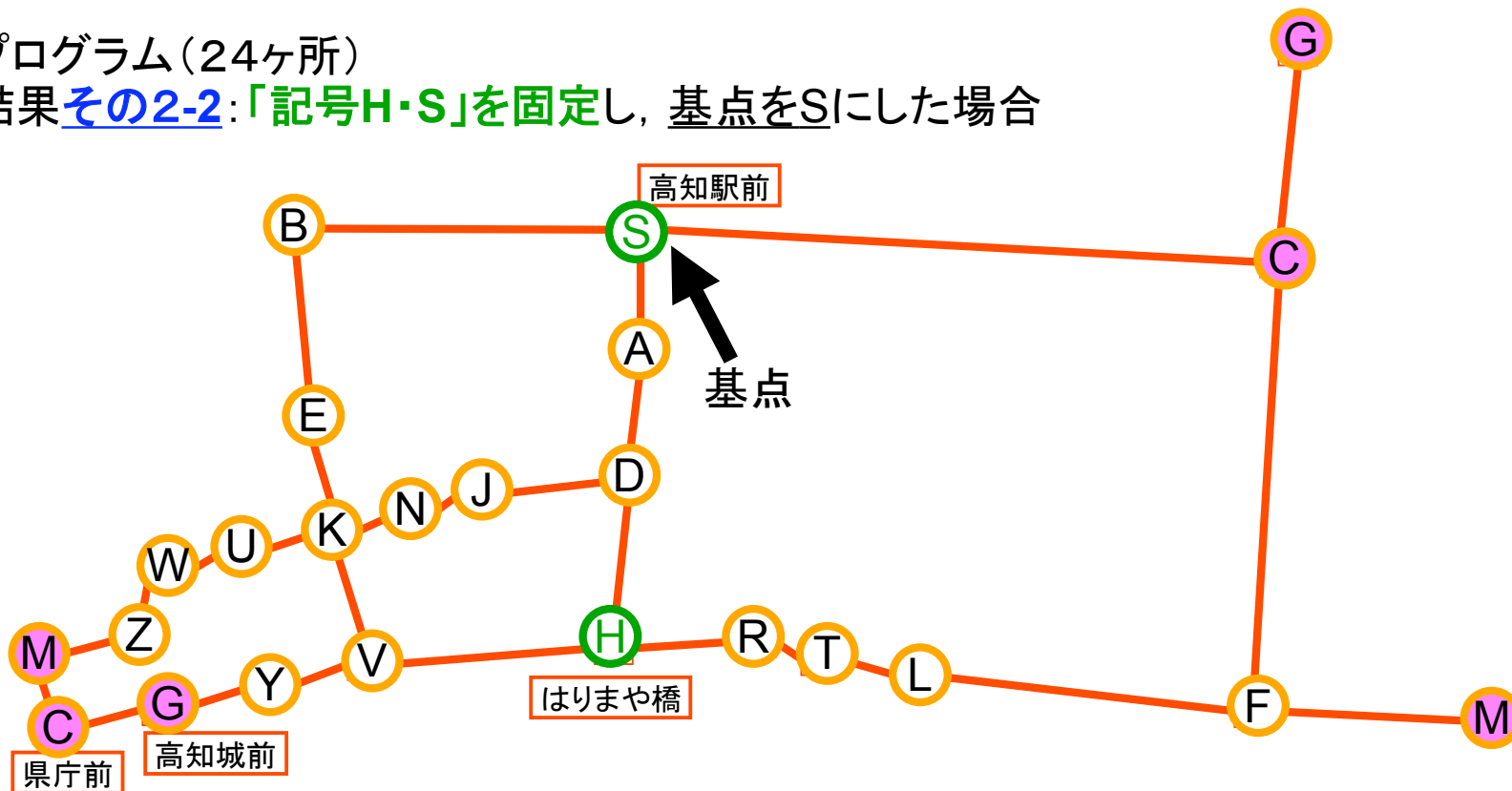
平均交差点数:9.0
平均距離:3.53

[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交	距離
T	2	13 20	9	3.999
V	2	4 14	9	4.323
W	2	12 19	9	2.26

配置プログラム(24ヶ所)

計算結果その2-2:「記号H・S」を固定し、基点をSにした場合

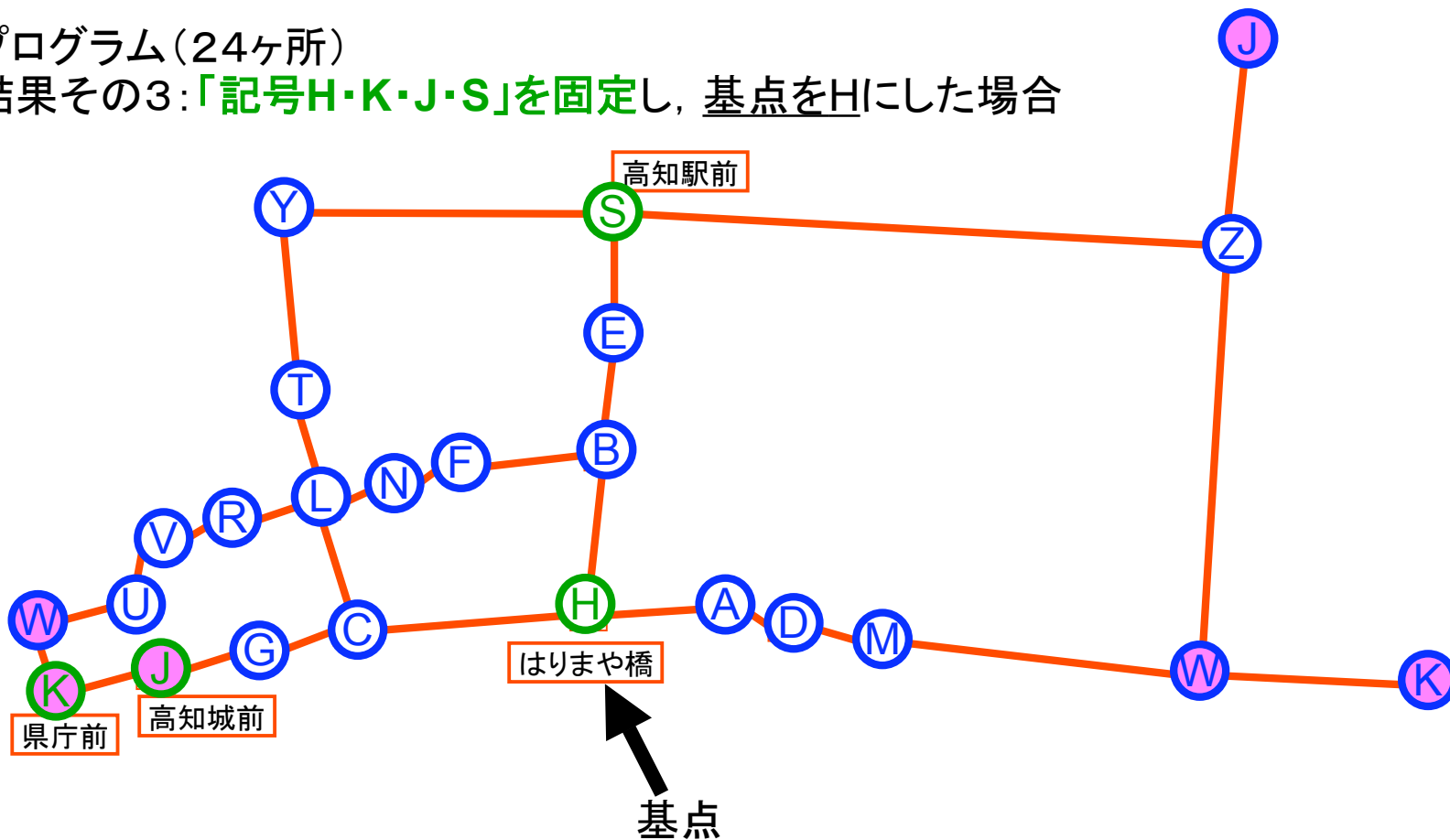


平均交差点数:8.7
平均距離:3.72

[重複]				
	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
C	2	4 15	8	2.849
G	2	5 14	8	4.134
M	2	13 19	10	4.183

配置プログラム(24ヶ所)

計算結果その3:「記号H・K・J・S」を固定し, 基点をHにした場合

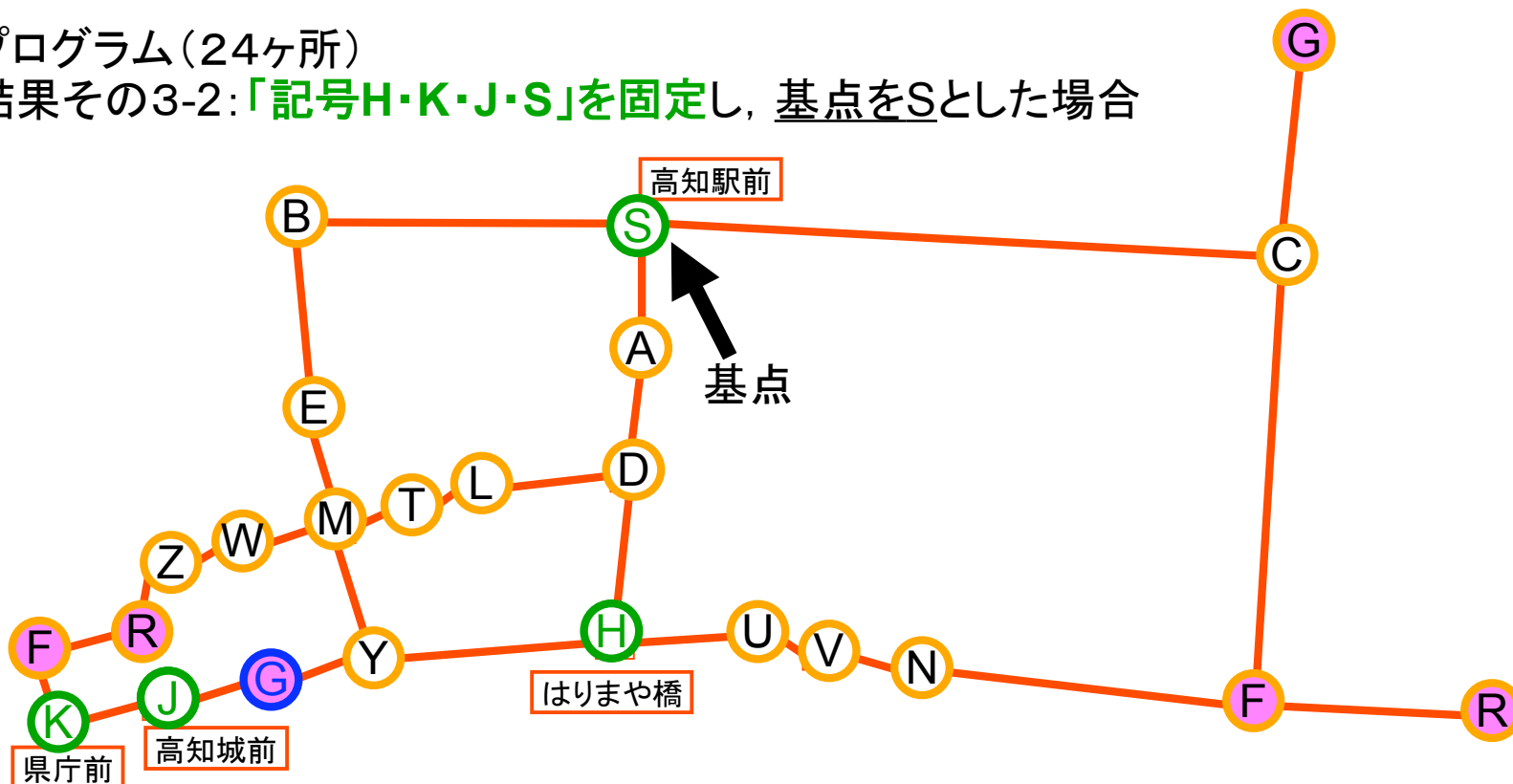


平均交差点数:8.7
平均距離:3.49

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
J	2	5 14	8	4.134
K	2	4 13	9	4.061
W	2	12 19	9	2.26

配置プログラム(24ヶ所)

計算結果その3-2:「記号H・K・J・S」を固定し, 基点をSとした場合

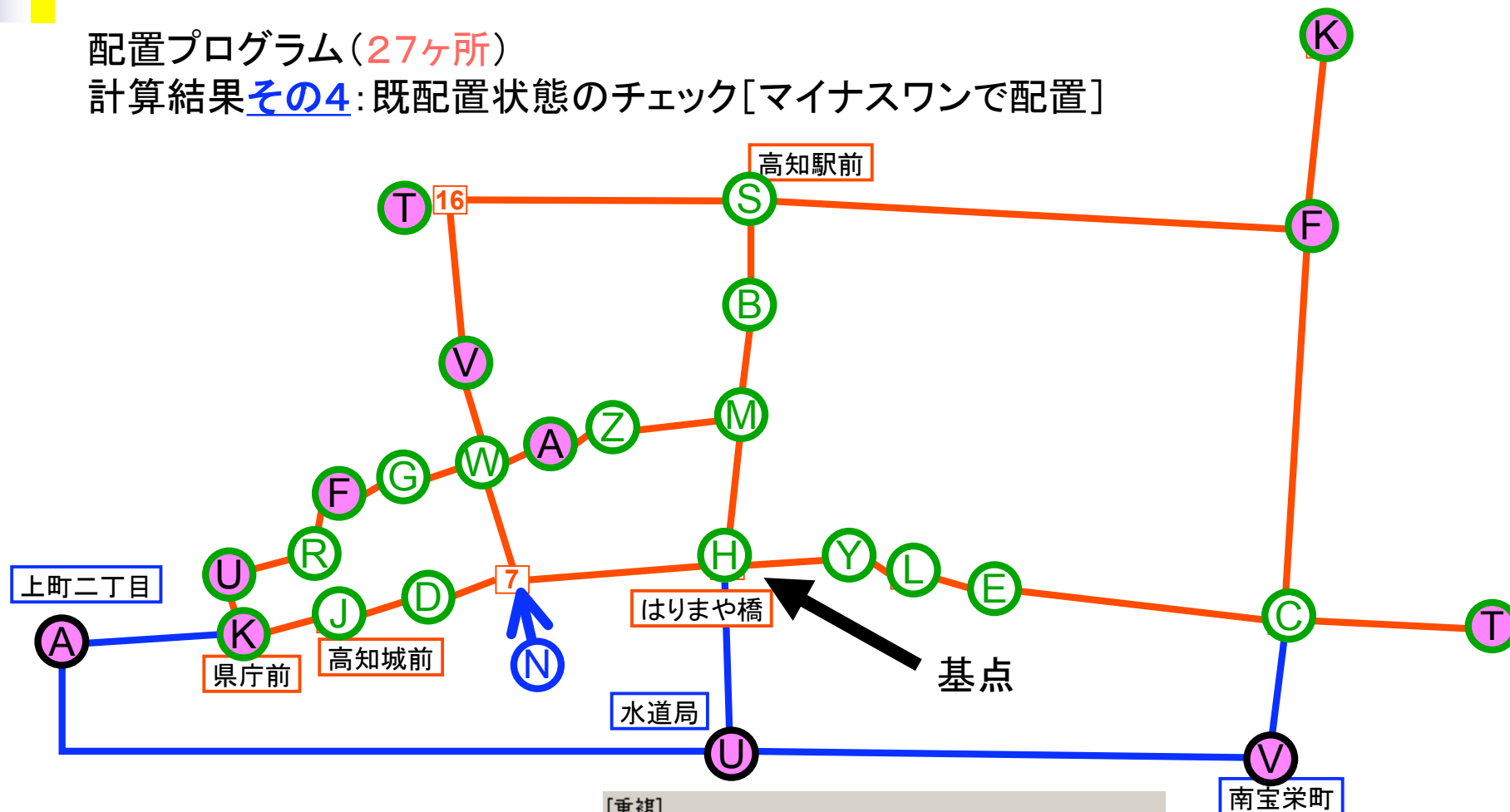


平均交差点数:8.3
平均距離:3.39

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
F	2	12 19	9	2.26
G	2	6 14	7	3.925
R	2	13 20	9	3.999

配置プログラム(27ヶ所)

計算結果その4: 既配置状態のチェック[マイナスワンで配置]



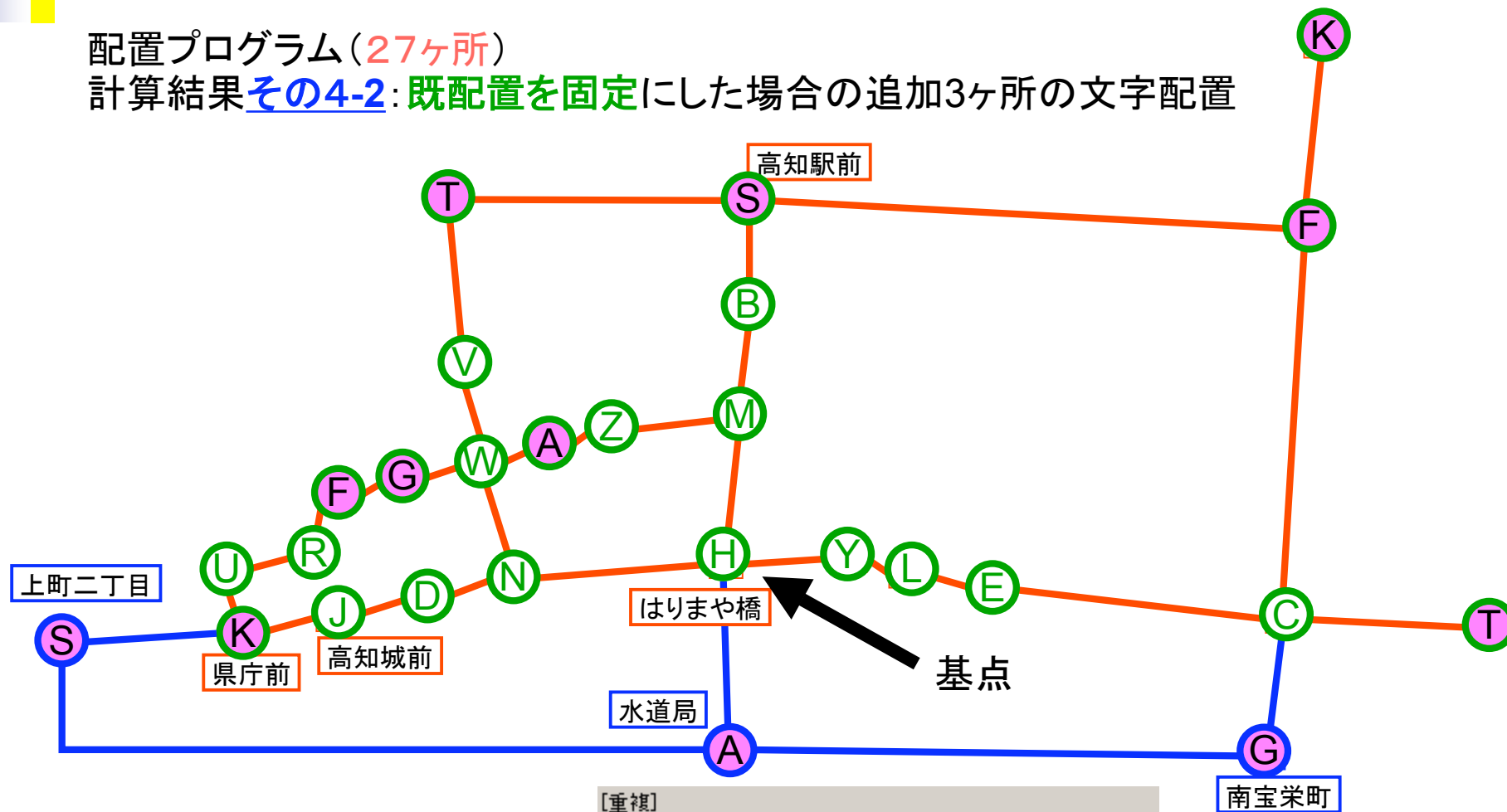
[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
A	2	23 25	4	2.613
F	2	15 21	6	2.376
K	2	4 14	6	4.323
T	2	13 16	4	4.267
U	2	19 26	3	1.181
V	2	17 27	4	2.222

平均交差点数: 4.5
平均距離: 2.83

配置プログラム(27ヶ所)

計算結果その4-2: 既配置を固定にした場合の追加3ヶ所の文字配置



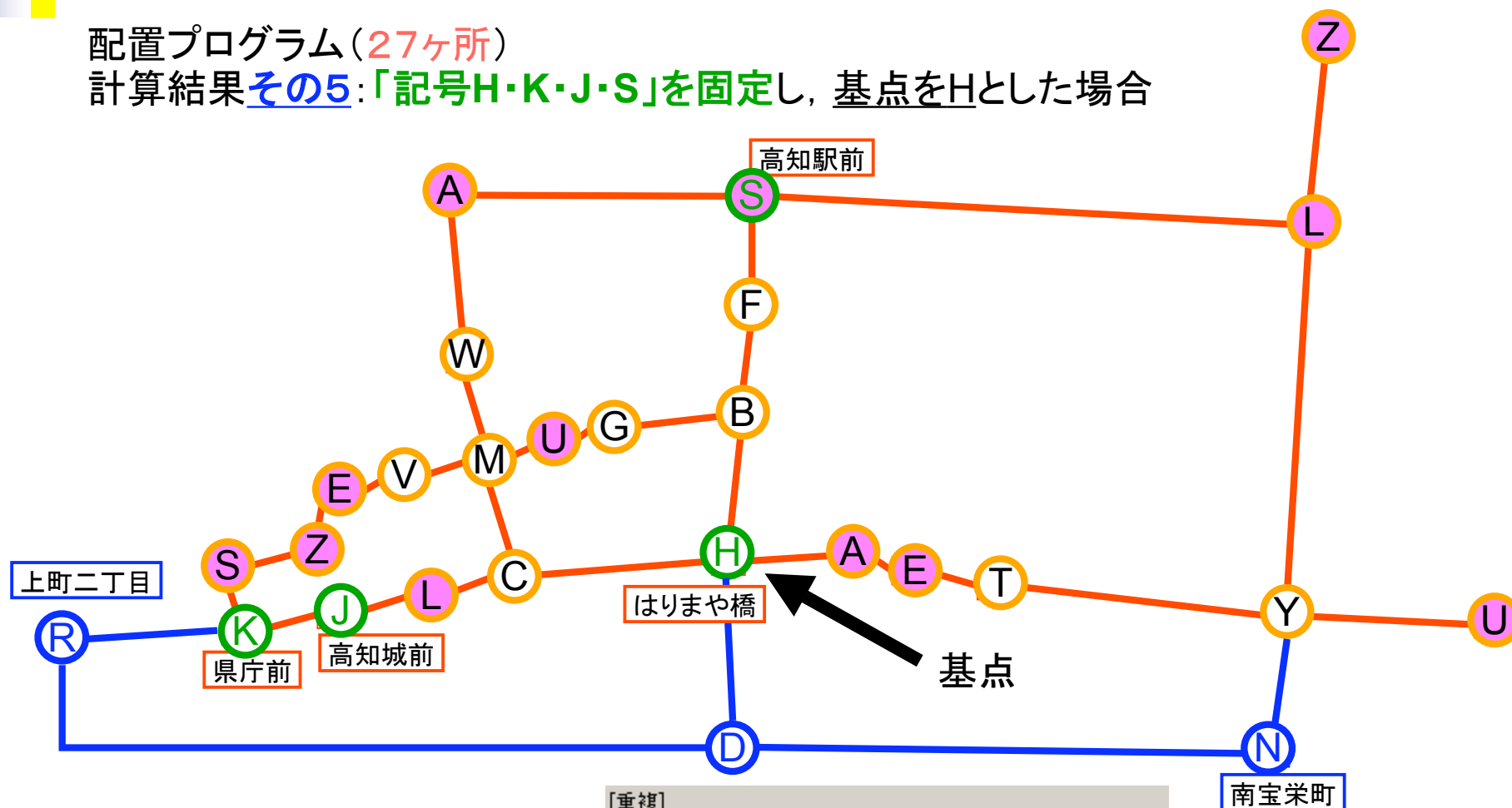
[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
A	2	23 26	4	1.626
F	2	15 21	6	2.376
G	2	22 27	5	2.207
K	2	4 14	6	4.323
S	2	1 25	2	2.579
T	2	13 16	4	4.267

平均交差点数: 4.5
平均距離: 2.90

配置プログラム(27ヶ所)

計算結果その5:「記号H・K・J・S」を固定し, 基点をHとした場合



[重複]

	同一記号使用交差点		最短道のり	
	個数	交差点	交...	距離
A	2	9 16	5	1.472
E	2	10 21	6	1.371
L	2	6 15	5	2.451
S	2	1 19	4	1.63
U	2	13 23	7	3.748
Z	2	14 20	7	4.039

平均交差点数:5.7
平均距離:2.45



配列プログラム(プロトタイプ)の作成

- ネットワーク情報(接続と長さ)を入力すれば**自動配列**できるプロトタイププログラムを準備
- **単純なアルゴリズム**により合理的な配列を実現
 - 最適化問題として解くのは困難なため近似解(→アルゴリズム高度化の余地)
 - アルゴリズムのキャリブレーションの余地はある
- **固定したい記号**は固定したまま配列可能
 - 「基点」付近ほど合理的に配列可能
- 配列の**合理性**を同記号間の最短経路上の「**交差点数**」と「**距離(長さ)**」などで**評価可能** <今は「交差点数」のみ>
- プロトタイプ・アルゴリズムなので**今後ほかの条件を加味**できる余地がある



配列プログラム(プロトタイプ)の作成

- プロトタイプ・アルゴリズムなので今後ほかの条件を加味できる余地がある

他の条件例

- 「記号化しない交差点」の数も考慮した配列
- 固定化する記号の優先度を加味した配列
(固定化したい記号と合理的配列の、整合性をとる配列)
- 直線上の同記号重複、ジグザグ上の重複、を区別した配列
- 隣り合う記号のペアが、重複しない配列
(Cの隣りがA,D,Sなら、他のCの隣りはA,D,S以外から選択する事で、どのCなのか認識しやすくする)



他配列方法との比較検証

■ 順列配列との比較検証

- 順列配列案(A)～国道をN、県道をK、市道をSとし、起点から順列で数字を振っていく(N-1,N-2,N-3,N-4,,,))
- 順列配列案(B)～各道路起点からAから順に振っていく(A,B,C,D,,,))
- 重複を許さない配列～国道をN、県道をK、市道をSとし、道路番号をふり、次に起点から順列で数字を振っていく(N32-1, N32-2, N32-3, N32-4,,,))
- 通り名方式～各通りに、通り名を振る。

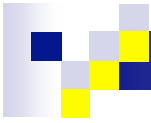
■ 将来の広域展開を視野に入れ、各案の検証を行う



他配列方法との比較検証

■ 順列配列案(A)の結果から分かる事

- 従道側交差点側の記号は順列にならず、順列が実現できる道路は国道の一部に限られる。
(国道の例:N-31 → N-32 → N-33 → N-34 → N-35,...)
(市道の例:S????-12 → S????-13 → K-2 → N-34 → K-25 → K-15 → S????-14,...)
- 道路の重複、分裂、道路起点の重複、などが多い場合、同記号、似た記号が集まってしまう。
- 順列中は、次の記号が推測できる効果がある。
- N-15とN-16の間の交差点を後で記号化する場合、N-15.1等、複雑な記号になる。
(※高速道路インターチェンジ番号における問題点と同様)
- 長い道路の場合、N-245など桁数が増え、ナビの注視時間が伸び、標識の判読距離が短くなると予想される。→交差点名の欠点が補完されない
- 隣合う記号は必ず似る(N-15とN-16等)ため、ナビの注視時間が伸び、標識の判読距離が短くなると予想される。→交差点名の欠点が補完されない
- K-15の隣りにK-15が並ぶなど、同記号が近くに配列される可能性が避けられず、ドライバーにとって見分けにくく混乱しやすくなる可能性がある。→交差点名の欠点が補完されない



他配列方法との比較検証

■ 順列配列案(B)～各道路起点からABCD...と

順に振っていく (Zの次はAに戻るルールで配列する)

(主道の記号を従道側でも表示するルールで配列する)





他配列方法との比較検証

■ 順列配列案(B)の結果から分かる事

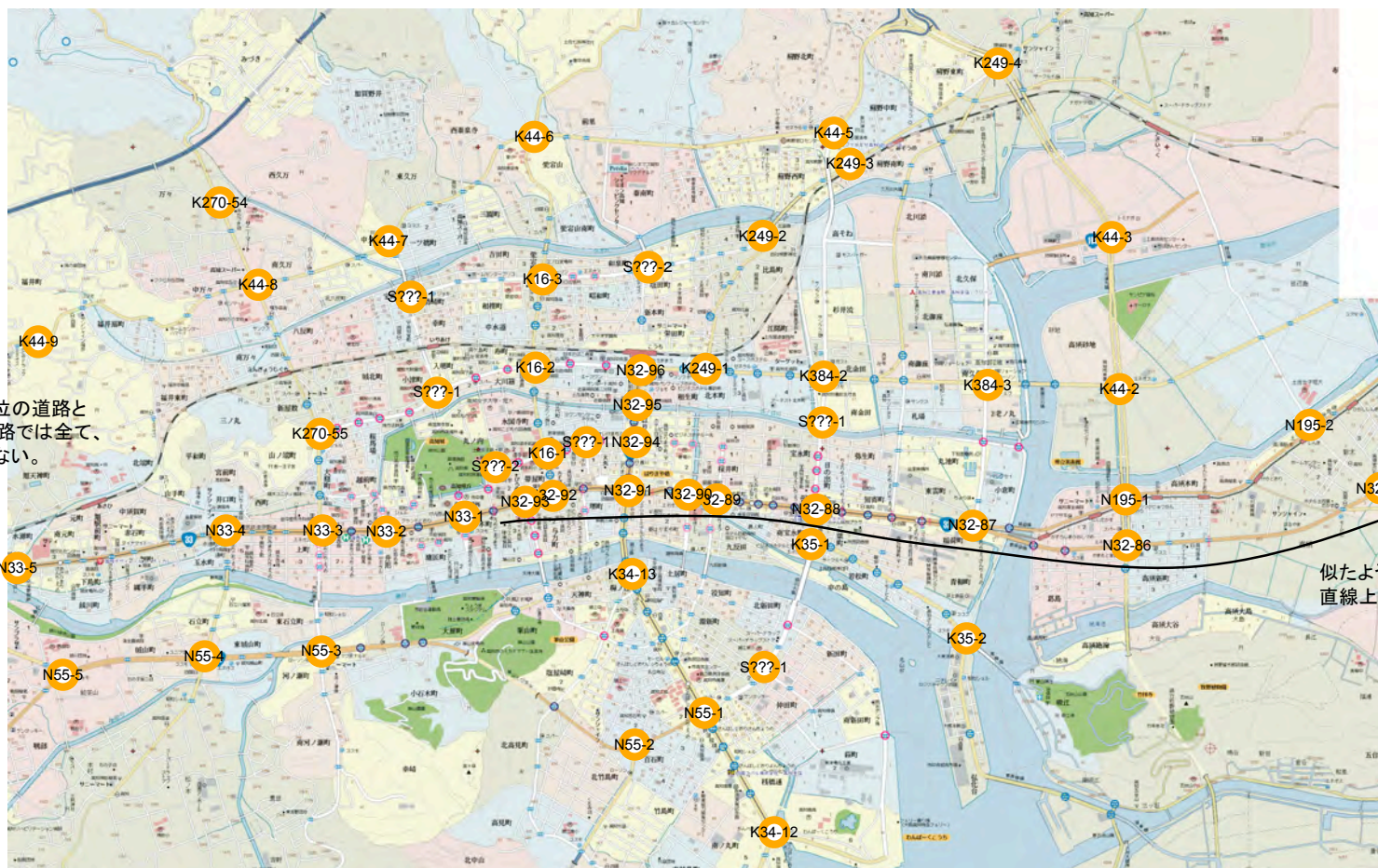
- 従道側交差点側の記号は順列にならないため、順列が実現できる道路は国道の一部に限られる。国道でも重複、分裂などある場合順列にならない。
- 順列上では次の記号が推測できる。他道路ではそうならない場合が多い。
(A → B → G → C → D → R → T → E → L → Z → F...)
- EとFの間の交差点を後で記号化する場合、順列とはならない。
- 同じ記号が隣り合ったり、近くに配列される事が避けられず、ドライバーが混乱しやすくなる可能性がある。



他配列方法との比較検証


■ 重複無し配列案

国道をN、県道をK、市道をSとし、道路番号をふり、次に起点から順列で数字を振っていく



自分より上位の道路と交差する道路では全て、順列にならない。

似たような長文の記号が直線上に並ぶ事になる



他配列方法との比較検証

■ 重複無し配列案の結果から分かる事

- 従道側交差点側の記号は順列にならず、順列が実現できる道路は国道の一部など、上位道路のない路線に限られる。
- 順列中なら、次の記号が推測できる効果がある。
- N32-15とN32-16の間の交差点を後で記号化する場合、N32-15.1等と複雑になる恐れがある。
(※高速道路ICの番号化における問題点と同様)
- N32-245など文字数が多いので、ナビの注視時間が伸び、標識の判読距離が短くなると予想される。
→交差点名の欠点が補完されない
- 同記号の重複による混乱はないが、隣合う記号は必ず似る(N32-15とN32-16等)ため、ナビの注視時間が伸び、標識の判読距離が短くなると予想される。
→交差点名の欠点が補完されない

他配列方法との比較検証

■ Ex. 信号制御器管理番号

交差点に置かれる信号機については地元の警察署が管理される。
信号機ごとの管理番号は、1信号付き交差点に1記号となるので、
交差点記号化にとっては参考例となる。

写真のK52-60の場合、

K=神奈川県警、50=都筑警察署、60=管轄内で60番目に設置された、という意味。

これは国内全(信号付)交差点を重複無く区別した例ともいえ、

重複無く全国を記号化する場合は、この記号のような長文になると分かる。

隣り合う交差点の番号は似る事が多く、判読しやすいとは言えない。



他配列方法との比較検証

■ Ex. 交差点名

現在日本国内の主要な交差点に置かれる「114-2Aおよび2B主要地点標識」は、元々ドライバーに同定性を与える(どこにいるか分かる)ために置かれたもの。

この標識が、カーナビなどの道案内の際に「曲がるべき交差点」の”特徴”として、「交差点名」と呼ばれ、広く活用されている。

この「交差点名」は、重複なく全国の交差点を記号化した例ともいえる。(偶然の重複は除く)

「霞ヶ関2丁目」「たまプラーザ駅前東側」など、長文になることが多い。また地名に依ると隣り合う名称同士が似る事が多く、間違えやすい。





他配列方法との比較検証

■ 通り名方式との比較検証

■ 例:パリ市>5E 区>@@通り>638

- 欧米では住居表示が通り名を基準に行われる。これを「道路方式」と呼びます。区までは面を区切り、@ @通りは線であり、その「線に沿った638番目の家」として住所が表記されるので、全ての道に通り名を付与する必要がある。そして通り名は交差「点」の角や信号機横に必ず掲示されている。

- ・通り名がどの通りにも付与されている
- ・通り名の考え方が住居表示を通じユーザーに浸透
- ・通り名が交差点部分に掲示されている



こういった土壌があって初めて、ドライバーへの道案内を、交差する「通り名」で行うルールが成立する。

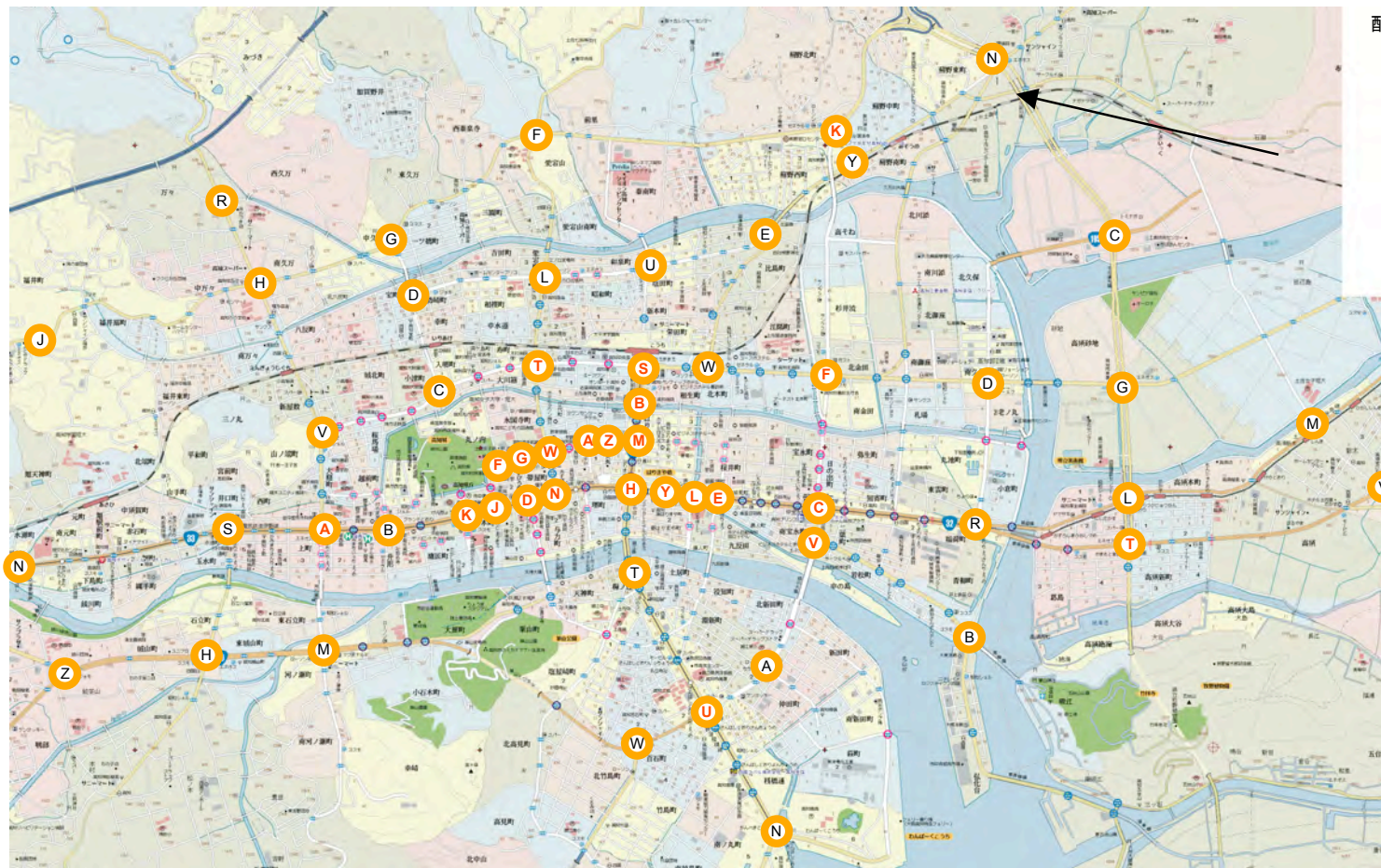
- ・全国に「通り名での道案内」を持込むには、住居表示も道路方式に改め、地元浸透させないといけない。
- ・歩行者観光客向けに、限定した街や商店街を通り名方式で運用するのは、街歩きには有効である。

- ・ドライバーにとっての「通り名の読みやすさ」は、「交差点名」とさほど変わらず、ナビや標識の注視時間、判読距離などの改善にはつながらない。



他配列方法との比較検証

■ 本提案と、他配列方法との比較



配列は仮のものです



他配列方法との比較検証

■ 配列結果からわかること

- 順列でないので、次の記号は推測できない。
- その代わりに、H=はりまや橋など、重要交差点では固定値を用い、交差点名に関連する1文字を配し、覚えやすさをサポートさせる事ができたりと、自由度がある。
- 1文字なので、ナビの注視時間、標識の判読時間が短い。
- 隣合う記号は決して似ないので、その事で混乱することがない。
- NとRの間の交差点を後で記号化する場合でも、そこから最遠の記号を配列できる。またこの繰り返りで、いくらでも後から記号化する事ができる。（その代わりに、同記号同士の距離は縮まる。）
- 特定の道路を順列で配列したい場合は、固定値をもちい、順列で配列する事も可能である。
(ただしPなど使用できない記号があるので、完全な順列とはなりにくい)



他配列方法との比較検証

■ 「順列でない事の混乱」また「同記号が重複する事の混乱」に関する、ナビ利用者アンケート結果からの考察

- ・ココ!マーク対応ナビをつかったレンタカー利用者100名へのアンケートで、「順列でない事の混乱」、「同記号が重複する事の混乱」の有無を尋ねた。
- ・「順列でない事の混乱」があった=24%、なかった=76%
- ・「同記号が重複する事の混乱」があった=17%、なかった=83%
- ・初めてココ!マークを利用するドライバーでも、その約8割の方が、混乱せず利用できた事がわかる。



他配列方法との比較検証

■ 「順列でない事の混乱」また「同記号が重複する事の混乱」に関する、ナビ利用者アンケート結果からの考察

・「混乱した」と回答した方の混乱が、深刻なものであったのか、その方の「ココ!マーク全体への評価」への回答態度から推測してみる。

全体のサンプル69人の平均 $(1*16+2*35+3*12+4*3+5*3)/69=2.16$ このときの分散=0.9595

質問2(1)45のどちらかで「混乱した」と回答した方=23名

質問3(3)7での回答内訳: (1が4名、2が14名、3が2名、4が1名、5が2名)

この人たちの平均値: $2.26...=(1*4+2*14+3*2+4*1+5*2)/23$ #このときの分散=1.2016

「質問2(1)の4)・5)どちらでも「混乱した」とは回答していない人=46人

その回答内訳:(1が12名、2が21名、3が10名、4が2名、5が1名)

よって、この人達の平均値: $2.11=(1*12+2*21+3*10+4*2+5*1)$ #このときの分散=0.8546

以上を用いて、質問2(1)の4)と5)への回答態度の違いによって、質問3(3)の7)への回答が異なるかどうかを統計的に確認結果、「両者が等しい」との仮説を捨てきれない確率が30%くらい。そこから、両者に差があるとは言えない(言い切れない)、と統計学からは帰結されます。

・「混乱した」と回答した方が、「混乱しなかった」方と比べ、さほど変わらない評価をしていたため、「混乱」した方の混乱は、さほど大きなものではなかったとも推測できる。



他地域での配列シミュレーション

- 客観基準により評価・確認する
- 交差点密度が最も高い地域での配列検証
- 分岐数が最も多い地域での配列検証

他地域での配列シミュレーション

■ 交差点密度が最も高い地域での配列検証

交差点の密集する地域と記号種類の関係 (種類数=25を優先)

最も交差点が密集している地域として、渋谷駅前と神田駅前を中心としたそれぞれ25主要な交差点の収まる面積を調べました。結果それぞれ3.2平方キロ、2.6平方キロの範囲に収まる事が分かりました。

[渋谷区道玄坂2丁目 \(渋谷駅前\)](#)

[千代田区神田須田町1丁目 \(神田駅前\)](#)

条件	交差点密集地域		
	神田駅前 主要な交差点	渋谷駅前 主要な交差点	
	交差点数	25	
	空間内の最多叉路	4叉路	4叉路
記号種類数	25		
結果	交差点の密度	2.6平方キロ/25記号	3.2平方キロ/25記号

■ 仮データです。ここは差し替えます



他地域での配列シミュレーション

■ 分岐数が最も多い地域での配列検証

■ 仮データです。ここは差し替えます

条件	空間	江戸川区中央3丁目内 全ての信号付き交差点 (4.7平方キロ)
	交差点数	05
	空間内の最多叉路	8叉路
	記号種類数	25
結果	記号の密度	1.6平方キロ/25 記号
	n=	5
	直線上でのn=	なし
	同記号同士の最短道のり 距離	1.2km



記号化標識の意義に関する再確認

主要地点標識(114-2Aおよび2B)との関係性

研究の最終目的は、114-2Aおよび2B主要地点標識と記号の一体化である。「114-2Aおよび2Bを交差点部に設置する場合には、記号も併記する」とした、標識令の改訂である。

↓
全国10万の主要交差点には、固有の交差点名+20記号のうちのどれかが、掲示される。
↓
ナビ、道路地図、108標識に記号を記載し、道案内時に活用する。
↓
ドライバは、今まで通り交差点名全てを読む場合と、記号部分のみ読む場合とを、必要に応じて使い分ける。

提案する記号標識の配置例 (114-2A標識の将来像)	
役割1	役割2
絶対位置目印  「交差点名」	相対位置目印  「記号」
目的地が「石神井公園」なら交差点名で確認する	経路途中の交差点なら「S」を用いる
停止し確実に把握する	運転中に覚え、見つけやすい

記号化標識の意義に関する再確認

主要地点標識(114-2Aおよび2B) との関係性

記号は、「ローソン」などのいわゆる交差点ランドマークと同じように使われる事が想定されている。

ローソンは全国に多くあるのに、道案内時に「ローソンを右」と広く活用されているのは、この交差点を全国唯一の地点として同定するのではなく、「周辺交差点との違い」で、交差点を説明したい時が多くあるから。運転中、いくつかの交差点のうちから曲がるべき交差点をパッとみつけない、といった場面がそれに当たる。

そういったニーズに合わせ、記号は1文字とし、聞いて覚えやすく、遠くから見つけやすいものとしている。また「いくつかの交差点」から比較する事が多いので、近隣に同記号がこないよう配列する。交差点名とも併記したかたちで、表示する。

- ・ナビや地図での道案内時には、記号で交差点を示し、ドライバーは記号を頼りに交差点を見つける。
- ・自分の位置を確実に同定したい場合には、停車して交差点名全部を判読する。

と使い分ける事を想定している。

提案する記号標識の配置例 (114-2A標識の将来像)	
役割1	役割2
絶対位置目印  「交差点名」	相対位置目印  「記号」
目的地が「石神井公園」なら 交差点名で確認する	経路途中の交差点なら 「S」を用いる
停止し確実に把握する	運転中に覚え、見つけやすい